

OCTOBER 1991

COMPUTER

Multimedia Information Systems

1951-1991



40 YEARS OF SERVICE
IEEE COMPUTER SOCIETY



THE INSTITUTE OF ELECTRICAL AND
ELECTRONICS ENGINEERS, INC.

Call For Papers

ASAP '92

**International Conference on
Applications-Specific Array Processors**

August 4-7, 1992

The Claremont Resort, Berkeley, California

Sponsored by the University of California at Berkeley in
cooperation with the IEEE Computer Society.

ASAP '92 brings together researchers, designers and users of applications-specific parallel computing from around the world. This is the sixth of the series starting with the First International Workshop on Systolic Arrays held in Oxford, England in 1986. Over the years, the scope has been expanding to reflect the growing interest in applications-specific computing systems.

All aspects of applications-specific parallel computing are eligible for ASAP '92, with emphasis on work which addresses both theoretical and practical implications. Topics of particular interest include:

- Applications
- Algorithms
- Design Methods/Tools
- Fault-tolerance
- Parallel Architectures
- Software Systems
- Modelling & Measurement
- Technology & VLSI/WSI
- CAD/CAM Frameworks

Submitted papers must indicate clearly how the work is relevant to applications-specific parallel computing. Reported results should be new and unpublished.

The conference will feature an opening keynote address, technical presentations, a panel discussion, poster displays, and a banquet. One of the poster sessions will be reserved for very recent results, ongoing projects and exploratory work. The site of ASAP '92 is The Claremont Resort in the hills above Berkeley, two miles from the UC Berkeley campus.

Authors are invited to submit five (5) copies of double spaced manuscripts, 5000 words or less in length, by **December 16, 1991**, to Mary Stewart, EECS Dept., Cory Hall, University of California, Berkeley, CA 94720. Notification of acceptance will be mailed by March 30, 1992. The camera-ready final paper is due by **May 1, 1992**.

Conference Chair:

José Fortes
Purdue University
fortes@ecn.purdue.edu

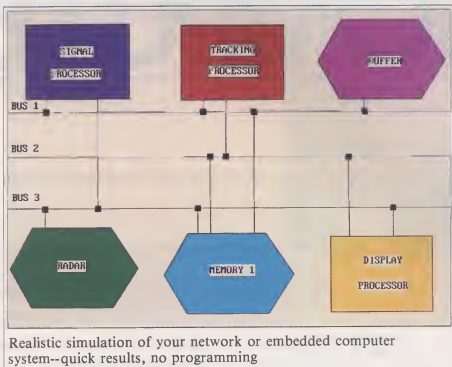
Program Chairs:

Edward Lee
UC Berkeley
eal@janus.berkeley.edu

Teresa Meng
Stanford University
meng@mojave.stanford.edu

Local Arrangements:

James Beck
International Computer
Science Institute
beck@icsi.berkeley.edu



NETWORK II.5 now predicts performance of computer-communication systems

Free trial and, if you act now, free training

NETWORK II.5 uses simulation to predict your network performance. You simply describe your network and workload.

Animated simulation follows immediately—no programming delays.

Easy-to-understand results

You get an animated picture of your network. System bottlenecks and changing levels of utilization are apparent.

Seeing your network animated increases everyone's understanding of its operation and builds confidence in your results.

Your reports show response times, messages delivered, messages lost, device utilization, and queueing statistics.

Computers with NETWORK II.5

NETWORK II.5 is available for most PC's, Workstations, and Mainframes.

Your network simulated

You can analyze embedded or distributed computer systems, or other computer-communication networks. Industry standard protocols such as FDDI and IEEE Standard 802.X are built-in. Others can be modeled.

You can easily study the effect of changing network parameters or even network protocols.

You can simulate some portions of the network at a detailed level and others at a coarser level.

Free trial information

The free trial contains everything you need to try NETWORK II.5® on your computer. For a limited time we also include free training —no cost, no obligation.

Call Paul Gorman at (619) 457-9681, Fax (619) 457-1184. In Europe, call Nigel McNamara, in the UK, on 0276 671 671, Fax 0276 670 677. In Canada, call Peter Holt on (613) 782-2474, Fax (613) 782-2202.

Free trial offer

☐ Yes I want to see how NETWORK II.5 quickly answers network performance questions.

Limited offer—Act now for free training.

Name	
Organization	
Address	
City	State Zip
Telephone	Fax
Computer	Op. Syst.

☐ Send details on your University Offer.

Return to: CACI Products Company
3344 North Torrey Pines Court
La Jolla, California 92037
Call Paul Gorman at (619) 457-9681
Fax (619) 457-1184
In Europe: CACI Products Division
Coliseum Business Centre
Watchmoor Park, Riverside Way
Camberley, Surrey GU15 3YL, UK
Call Nigel McNamara on 0276 671 671
Fax 0276 670 677
In Canada: CACI Products Company
200-440 Laurier Avenue West
Ottawa, Ontario K1R 7X6
Call Peter Holt on (613) 782-2474
Fax (613) 782-2202

COMPUTER

October 1991 • Vol. 24, No. 10

ARTICLES

6 Multimedia Information Systems: The Unfolding of a Reality

Guest Editors' Introduction: A. Desai Narasimhalu and Stavros Christodoulakis

9 Advances in Interactive Digital Multimedia Systems

Edward A. Fox

Progress in hardware, algorithms, and standards has facilitated interactive access to digital multimedia information. This tutorial introduces basic concepts in digital multimedia systems and surveys recent literature.

23 Conceptual Modeling of Multimedia Documents

Carlo Meghini, Fausto Rabitti, and Costantino Thanos

Multos is an experimental system designed to enhance multimedia document retrieval. It is based on multilevel conceptual modeling principles for semantic structure.

33 Automating the Generation of Coordinated Multimedia Explanations

Steven K. Feiner and Kathleen R. McKeown

We developed an explanation system to overcome the disadvantages of conventional authoring in multimedia applications. COMET not only determines what to say, but how to say it.

42 Spatio-Temporal Composition of Distributed Multimedia Objects for Value-Added Networks

Thomas D.C. Little and Arif Ghafoor

Future multimedia information systems will require the integration of various objects retrieved from databases distributed across the network. Here we discuss a methodology for spatial and temporal composition of multimedia objects.

51 A Continuous Media I/O Server and Its Synchronization Mechanism

David P. Anderson and George Homsy

Acme is a network server for digital audio and video I/O. It lets users specify their synchronization requirements through an abstraction called a logical time system.

58 InternetExpress: An Inter-Desktop Multimedia Data-Transfer Service

Murugappan Palaniappan and George Fitzmaurice

Featuring a simple desktop operation that works uniformly across all applications, the InternetExpress integrated service enables users to send and receive selection-based multimedia data in a timely manner.

69 Multimedia Conferencing in the Etherphone Environment

Herrick M. Vin, Polle T. Zellweger, Daniel C. Swinehart, and P. Venkat Rangan

The latest extension of the Etherphone project creates a powerful conferencing system that lets users control their participation in multiple conferences across multimedia networks.

Published by the IEEE Computer Society

Circulation: *Computer* (ISSN 0018-9162) is published monthly by the IEEE Computer Society, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1264; phone (714) 821-8380. IEEE Computer Society Headquarters, 1730 Massachusetts Ave., NW, Washington, DC 20036-1903; IEEE Headquarters, 345 East 47th St., New York, NY 10017. Annual subscription included in society member dues. Nonmember subscription rates: available upon request. Single copy prices: members \$10.00; nonmembers \$20.00. This magazine is also available in microfiche form.

Postmaster: Send undelivered copies and address changes to *Computer*, IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08855. Second class postage is paid at New York, New York, and at additional mailing offices.

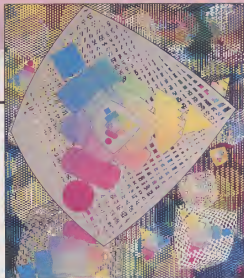
Copyright and reprint permission: Copyright © 1991 by the Institute of Electrical and Electronics Engineers, Inc. All rights reserved. Abstracting is permitted with credit to the source. Libraries are permitted to photocopy beyond the limits of US copyright law for private use of patrons: (1) those post-1977 articles that carry a code at the bottom of the first page, provided the per-copy fee indicated in the code is paid through the Copyright Clearance Center, 28 Congress Street, Salem, MA 01970; (2) pre-1978 articles without fee. Instructors are permitted to photocopy isolated articles for non-commercial classroom use without fee. For other copying, reprint, or republication permission, write to Permissions Editor, Computer, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1264.

Editorial: Unless otherwise stated, bylined articles, as well as product and service descriptions, reflect the author's or firm's opinion. Inclusion in *Computer* does not necessarily constitute endorsement by the IEEE or the Computer Society. All submissions are subject to editing for style, clarity, and space considerations.



DEPARTMENTS

- 4 President's Message
Growing up
- 81 Product Reviews
Canon printers
- 88 New Products
- 90 IC Announcements
- 92 Microsystem Announcements
- 94 Update
- 95 Computer Society News
- 96 Standards
- 104 Call for Papers
- 106 Calendar
- 110 Book Reviews
- 112 Open Channel



Cover: Matthew Shriver

In the next issue

Encryption, operating systems, molecular biology, medical monitoring, corporate strategies for information technology, and computer curricula

Career Opportunities, 98; Computer Society Information, 103; Membership Application, 97; Change of Address Form, 96; Advertiser/Product Index, 80; Reader Service Card, 80A

COMPUTER

10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1264

Editor-in-Chief

Jon T. Butler, Naval Postgraduate School

Editorial Board

Dennis R. Allison, Stanford University
Laxmi N. Bhuyan, Texas A&M University
Thomas A. Corbi, IBM
Scott Davidson, AT&T
Richard Eckhouse, MOCO
Michael Evangelist, Andersen Consulting
Scott E. Fahman, Carnegie Mellon University
John R. Gurd, University of Manchester
Brent T. Halpern, IBM T.J. Watson Research Center
Tadao Ichikawa, Hiroshima University
Guy Johnson, Rochester Institute of Technology
Alan Kaminsky, Rochester Institute of Technology
Yale Patt, University of Michigan
Guyline M. Pollock, Sandia National Laboratories
Vincent Y. Shen, Hong Kong University of Science and Technology
Martha Sloan, Michigan Technological University
Will Tracz, IBM
Akinori Yonezawa, Tokyo Institute of Technology

Submissions

Send 12 double-spaced copies of articles and special-issue proposals to Jon T. Butler, Dept. of Electrical and Computer Engineering, Code EC/BU, Naval Postgraduate School, Monterey, CA 93943-5004, phone (408) 646-3299. For electronic submission, Butler's addresses are j.butler on Compmail II and butler@ecc.nps.navy.mil on Internet.

Staff

Managing Editor: Marilyn Potes
Staff Editors: Christine Miller
Chuck Governale

Bob Carlson
Linda World

Contributing Editor: Ware Myers

Art Director: Jay Simpson
Design: Toni Van Buskirk

Production: Chris Patterson, Ken Duckworth

Publisher: True Seaborn

Editorial Director, CS Magazines: Marilyn Potes

Assistant Publisher: Douglas L. Combs

Assistant to the Publisher: Pat Paulsen

Membership/Circulation: Christina Champion

Advertising Coordinators: Heidi Rex, Marian Tibayan

Magazine Advisory Committee

James J. Farrell III (chair), Fiorenza Albert-Howard, Valdis Berzins, Jon T. Butler, B. Chandrasekaran, Carl Chang, Manuel D'Abreu, Dante Del Corso, Gerald Engel, Michael Evangelist, James J. Farrell III, Tse-yun Feng, Anil K. Jain, Ted Lewis, Ray Miller, Michael C. Mulder, C.V. Ramamoorthy, True Seaborn, Sallie Sheppard, Harold Stone, Pradeep K. Srimani, Peter R. Wilson

Publications Board

Ronald G. Hoelzeman (chair), James Aylor, Victor R. Basili, Richard Burke, Jon T. Butler, J.T. Cain, B. Chandrasekaran, Carl Chang, Manuel D'Abreu, Dante Del Corso, Gerald Engel, Michael Evangelist, James J. Farrell III, Tse-yun Feng, Anil K. Jain, Ted Lewis, Ray Miller, Michael C. Mulder, C.V. Ramamoorthy, True Seaborn, Sallie Sheppard, Harold Stone, Earl E. Swartzlander, Peter R. Wilson



Growing up

There are three major issues I would like to discuss over the next three months. One deals with our identity: Who are we, and who should we be? We have been giving this a lot of thought lately, relative to the growth of the society and the services we offer, and to whom we recruit and retain. We will, no doubt, continue this self-examination well into next year. Another issue deals with what I regard as our primary function: providing our members with the information they need in a form that is useful to them. We must move ahead in this area, exploring different technologies for information dissemination. Hopefully, I will have more to say about this later this year. And finally, there is the issue of growth and size and the balance between volunteers and staff, which I will discuss this month.

As anyone who has started a new company can tell you, there are significant growing pains as a small company becomes larger. Things once done in ad hoc ways become more regimented as the number of tasks grows beyond what a few individuals can do by themselves. This regimentation is necessary to ensure consistency as people move in and out of the organization. Often it is perceived as bureaucratic; in fact, it is a necessary part of the growth that allows the organization to move forward without reexamining each and every action. This is not unlike humans growing up — rather than thinking about every action, we formulate certain rules and principles that allow us to get on with life and free up time to think about nonroutine events.

A professional society is no different, but has perhaps an added dimension — staff and volunteer balance. Most professional societies are directed by volunteers, that is, members of the society whose primary job is a professional one, but who volunteer time to the affairs of the society. As the society grows, more of the work is turned over to the staff. This is not just inevitable; it is good. It ensures consistency, since volunteers tend to come and go more frequently than staff. Further, while the volunteers may well be professionals in terms of their own jobs, they usually are not professionals when it comes to running a professional society. In the best of worlds, the volunteers do what they can do best, chart directions and major policies, while the staff has the operational authority and the responsibility to advise volunteers in policy issues.

Ed Parrish, who had concentrated his volunteer work outside the society for several years, came back to us as president-elect in 1987. I remember him telling me that he was literally stunned at the changes in our operations during the approximately six years he had been away. It's been only three years since Ed's term as president ended (he currently serves as an IEEE division director), but I'd bet he'd be almost as surprised now. It has become trite, though no less true, to speak of the rate of change in our technology and in the societies in which we live and work. Your Computer Society reflects that change as it keeps pace with technology, society, and the needs of its members.

No longer a mom-and-pop small business, we're now a substantial concern. Still not on a par with large corporations, in business terms we are a pretty large small business. Our budget exceeds \$20 million a year. We publish seven magazines, five transactions, and operate a press that will put out more than 100 books this year. We organize, sponsor, and cosponsor conferences that literally span the globe, and our more than 200 standards groups are having a tremendous effect on the industry worldwide. We are active in educational curricula development and minority education issues. Our membership of computer professionals is the largest in the world, and we are the largest society in the IEEE, accounting for approximately one third of its total membership and one third of its publications. On top of all this, the complexity of the environment in which we work (financial, legal, regulatory, technological), especially with offices in three different countries, means that nothing is simple anymore.

One important change from the old days, crucial to our current and future success, has been the expansion and increased professionalism of our staff. In 1986, under then-president Roy Russo, we began to emphasize staff-volunteer partnerships in most of our program areas. Our staff are not just "hired hands" who carry out volunteer instructions. They are engaged professionals who share our volunteers' commitment to providing quality services to the membership in the most effective and efficient ways possible. I can honestly say that the opportunity to work closely with them is one of the special rewards of volunteer service to the society. As a group they are conscientious,

hard-working, and exhibit a commitment to quality all-too-unusual these days. They provide a leveraging for volunteer effort that lets us be much more productive and effective than was previously possible. Sometimes we don't make it easy for them. We volunteers establish policies and rules, which our staff then enforces. It's obviously unfair for those who object to the policy or rule to blame the staff as "bureaucratic," but it happens.

Our staff now totals about 90 individuals, but we remain leanly staffed when compared to other organizations of our type and size and to our parent, the IEEE. We devote about 18 percent of our total budget to the salaries and fringe benefits of our employees — a statistic that has been virtually constant for several years. Available data suggest that the "typical" large professional association expends 26 percent of its funds on staff compensation.

The job of president of this society is not just honorary. It is a working job, and for those of us with regular "day jobs," being president would be impossible were it not for one of the most critical changes in the recent history of the society. In 1981, recognizing that our growth mandated more "hands on" leadership, a decision was made under then-president Oscar Garcia to create the position of executive director. T. Michael Elliott was recruited to fill that post in 1982 and continues to serve us in that capacity.

As expected, our staff has continued to grow, largely as a result of program expansion: We've more than doubled the number of periodicals published, books produced, and conferences sponsored in that time. In addition, we've opened European and Asian offices to help us better serve our non-US members. But this growth has not been in numbers alone. It is the growth in professionalism of the staff, from editors to accountants, coordinators to clerks, that is so impressive.

Quality may be the buzzword of the nineties, but Michael Elliott has had a quality management program in place since 1986, with results apparent to all of us who interact with the staff. The budgeting techniques he devised are now models for other organizations, including the IEEE. The Computer Society was recently recognized as one of 20 "successful" associations in a study and book published by the American Association of Society Executives intended to help other organizations emulate our success as a market-driven association, committed to staying in close contact with our membership (see *Computer Society News*, p. 95). Michael would be the first to agree that we're not perfect. But we're pretty good, and the reason we are is that we keep trying to get better through training, reviews, planning, and feedback. My recent predecessors and I especially appreciate his efforts to keep important policy decisions open for volunteer determination. He necessarily makes operational decisions every business day, but policy matters, those that concern the fundamental nature of the society, its services, and its future direction, he carefully frames in a manner conducive to good decision making. That includes presenting as much good information and data as he can on which we might base a decision.

We are growing up. We are a healthy organization with, I believe, a good staff-volunteer balance. But while we have introduced a certain level of regimentation that permits us to function efficiently, periodically we must still re-examine what we are doing and adapt to a changing environment. One such reexamination is currently being completed by my successor, Bruce Shriver. You will hear more about this effort in coming months.

Duncan H. Lawrie
Computer Society president

Plenum Books VITAL RESOURCES IN COMPUTER SCIENCE

Forthcoming . . .

ASSOCIATIVE COMPUTING A Programming Paradigm for Massively Parallel Computers

by Jerry L. Potter

Integrating associative processing concepts with massively parallel SIMD technology, *Associative Computing* explores a model for accessing data by content or meaning rather than abstract address mapping. Selected chapters introduce associative computing and the Single Tabular Data Associative Computation (SITDAC) model, describe Associative Computing (ASC) programs, illustrate context-sensitive compilation, and discuss the implementation of a Prolog interpreter in ASC. A volume in the series *Frontiers of Computer Science*.

0-306-43987-5/approx. 350 pp./ill./1992

SOFTWARE REUSE Guidelines and Methods

by James W. Hooper and Rowena O. Chester

This useful and well-written volume summarizes the latest research and practice in software reuse and provides practical guidelines for integrating software reuse into a software engineering process. Commencing with a brief background of software reuse, Hooper and Chester provide an in-depth look at managerial and technical guidelines, including financial and legal considerations, organizational management and structure, and tools and environments. A volume in the series *Software Science and Engineering*.

0-306-43918-2/196 pp./ill./1991/\$45.00

FRONTIERS OF COMPUTING SYSTEMS RESEARCH

Series Editor: S. K. Tewksbury

ESSAYS ON EMERGING TECHNOLOGIES, ARCHITECTURES, AND THEORIES

edited by S. K. Tewksbury

Forthcoming . . .

Volume 2

Selected chapters in the second volume include • limits of information processing systems • optics in a multiple-instruction, multiple-data stream computer • clouds, computers, and complexity.

0-306-44032-6/approx. 400 pp./ill./1991

Volume 1

Selected chapters in the first volume in the series include • stability of continuous cellular automata • paradigms for optical computing • unusual graphic representations of complex data • synchronization in hard real-time systems.

0-306-43590-X/400 pp./ill./1990/\$65.00

HUMAN-MACHINE INTERACTIVE SYSTEMS edited by Allen Kilnager

Examining recent and potential developments in display techniques, technologies, and algorithms, this comprehensive volume covers the latest data on knowledge, information, and communication in computing. A volume in the series *Languages and Information Systems*.

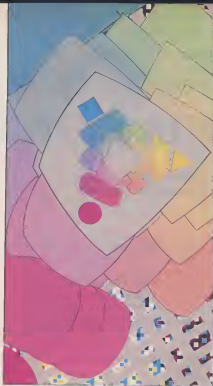
0-306-43758-9/325 pp. + index/ill./1991/\$79.50

Book prices are 20% higher outside US & Canada.

PLENUM PUBLISHING CORPORATION

233 Spring Street, New York, NY 10013-1578

Telephone orders: 212-620-6000/1-800-221-9399



GUEST EDITORS' INTRODUCTION

Multimedia Information Systems: The Unfolding of a Reality

A. Desai Narasimhalu, National University of Singapore
Stavros Christodoulakis, Technical University of Crete

The promise of multimedia information systems is just now being fulfilled. Facsimile transmission of black-and-white images is common in offices and is rapidly gaining acceptance in homes. Some fax machines transmit color images as well. Many offices are implementing document image-handling systems that integrate images into office text work flow. The combination of document image-handling systems, fax, and e-mail technologies can significantly improve productivity.

CD-ROM technology has matured, making many titles available and reducing production costs for discs and CD-ROM drives. Digital video interactive technology, which supports full-motion digital video on computers, may one day rival the resolution of high-definition television. Communications carriers around the world are deploying optical fibers, offering the high communication bandwidth needed for multimedia systems.

Multimedia technology is distinguished from existing technologies like television by the interactivity it gives users; hence, the term multimedia information systems. In this special issue, we outline the status of these systems, describe some technologies and current implementations, and discuss their fu-

ture. The sidebar presents acronyms common in multimedia technology and used throughout this issue.

What else is in store?

Recent announcements in storage technology, such as Nippon Telegraph and Telephone's one-terabyte rewritable magneto-thermal optical discs, are paving the way for a new range of applications. Jukeboxes using these discs, along with image retrieval systems such as Kodak's Photo CD system, will make possible large multimedia repositories. Such repositories contain objects in electronic form for on-line access. Examples include catalogs of museum collections, music and video archives, newspaper clips, and photo banks. Newspaper and magazine publishers, broadcast stations, museums, national archives, and architects are potential users of such repositories. Information kiosks, video teleconferencing, and interactive education tools are potential applications.

Three-dimensional sound technology, which can generate a physical representation of a sound or piece of music in space within two degrees of accuracy, promises more realistic audio simulations. The rapid progress in artificial

realities has vastly improved the "look and feel" of interfaces and provided sophisticated interactive simulated environments for a variety of applications that can benefit from multimedia repositories.

The industry response

Three years ago, Microsoft presented specifications for a multimedia PC at its CD-ROM and Multimedia Conference. These specifications represented the first significant attempt to define a hardware platform for multimedia applications. At least 20 different vendors have since committed themselves to MPC platforms. This is a clear sign that vendors see serious potential for multimedia applications. Proponents claim that by the late 1990s, such MPCs will have four "G's": a gigabyte of main memory, a gigabyte (at least) of secondary storage, giga operations per second, and gigabit-per-second data-transfer rates. The recently announced Fujitsu FM-Towns and Commodore Dynamic Total Vision systems both satisfy the MPC specifications.

But the big industry debate is whether multimedia technology will gain wider user acceptance through the consumer electronics or the computer market.

While CD-ROM and other products are introducing multimedia technology into the computer market, CD-I technology from Philips and other announcements from Sony are introducing multimedia technology through the consumer electronics market. Initial indications are that multimedia technology will gain greater market acceptance through consumer electronics. The recent announcement that Nintendo is embracing CD-I technology adds strength to this view. Indeed, about 35 percent of American homes have a Nintendo machine.

Potential impact

With the impending introduction of moderately priced digital video editing workstations (either CD-I or DVI), the advertising industry may experience a major change in the way it does business. Such systems will allow free-lance agencies to produce inexpensive rough cuts and prepress clips, thereby reducing the dominance of the larger agencies in this sector. An expected drop in the prices of 3D graphics workstations will support exciting new opportunities to combine digital video and 3D graphics in advertising.

The engineering industry will benefit from multimedia technology as well. Given the progress in multimedia technologies and object-oriented database systems, in a few years architects and designers will be able to use these systems to call up typical designs from a multimedia repository, customize the design to suit a job on hand, visualize the design in 3D, make a video clip, and send the clip to a client. The power to simulate high-resolution 3D designs will speed up design cycles. Add to this the possibility of using expert systems that can pinpoint potential problems in a production line, and the prospect emerges for reducing "concept-to-production" cycle time. This can benefit manufacturing and enable product customization at less cost.

A third area where multimedia technologies will make an impact is education. Recall the math and language machines from Texas Instruments many years ago; then envision the proliferation of handheld multimedia machines for the study of chemistry, physics, and math. If these machines prove successful, they could lead to some rethinking of educational models and the roles of

schools and teachers. For example, schools may abandon the cohort progress of all students at the same pace, and teachers may be freed to more fully assume the role of facilitators rather than knowledge-transferring agents.

Concerns regarding standards

Standards are still a major issue in multimedia information systems. There has been some convergence. For example, industry has agreed to the JPEG standard for still-image compression and the MPEG standard for moving-image compression. The High Sierra group's efforts have led to standards for some families of optical discs. More than 17 vendors, including Microsoft, have adopted the MPC concept, which appears to be a move in the right direction.

Window management systems have not yet converged; Microsoft's Windows 3.0, Sun's News, and Massachusetts Institute of Technology's X Windows

are the strong contenders. No agreement has been reached on standards in the "look and feel" of these systems (that is, the way such interface features as buttons and menus are presented).

A number of standards committees are looking at the different layers of technologies in a multimedia system, and their efforts are crucial to bringing about convergence.

Application-development methodology

The absence of a well-defined application-development methodology is a major restraint on the growth of multimedia technology. Too many teams from traditional software development houses are finding little success in developing multimedia applications with conventional methodologies. Multimedia application development involves teams of people who have different problem-solving approaches. For example, computer professionals tend to think in very

Acronyms

ADPCM	Adaptive differential pulse-code modulation
AVK	Audio/video kernel
CD-I	Compact disc interactive
CD-ROM	Compact disc read-only memory
CIF	Common intermediate format
CLUT	Color look-up table
Codec	Coder/decoder
DCT	Discrete cosine transformation
DSM	Digital storage medium
DSP	Digital signal processing
DVI	Digital video interactive
FDCT	Forward discrete cosine transformation
HVS	Human visual system
IDCT	Inverse discrete cosine transformation
IMCS	Interactive multimedia computing systems
JPEG	Joint Photographic Experts Group
MHEG	Multimedia and Hypermedia Information Coding Expert Group
MPC	Multimedia personal computer
MPEG	Moving Picture Coding Experts Group
ODA	Office document architecture
PAL	Phase alternating line
PCM	Pulse-code modulation
QCIF	Quarter common intermediate format
SGML	Standard generalized markup language
SMPTE	Society of Motion Picture and Television Engineers
VQ	Vector quantization
WORM	Write once, read many

structured ways, whereas professionals from creative services, such as audio and video, tend to have either an unstructured or semistructured approach to their projects. The differences in their training and approaches to development will pose a major challenge to the project managers of such development teams. It is therefore important for a good multimedia application development methodology to be identified and popularized soon.

What are the research challenges?

Multimedia information systems will have very large storage requirements. Some of the stored data will be read-only, some, write-once, and some, erasable. The challenge lies in designing efficient data-management hierarchies involving all these storage technologies and providing reasonable response times for access to present and future data collections. Such optimizations may use parallelism, cache memory, migration algorithms, and scheduling.

Real-time multimedia data-transfer and synchronization algorithms will be another important area of research. Research on similarity measures for text, speech, graphics, and images will play an important part in content-based retrieval techniques for multimedia information systems. Relevance feedback and incremental query refinement techniques will make these systems more usable. The research in virtual realities that addresses the "look and feel" of user interfaces will improve the quality of interaction with multimedia information systems.

We can expect multimedia systems to become increasingly important in our homes and offices. The technical challenges are more than offset by the broader means of communication and the more interactive mode of understanding. Because the communication is broader, the development of multimedia information systems will require innovative development approaches. The articles in this issue of *Computer* address some of the challenges and approaches to this exciting technology. ■

Acknowledgments

We thank everyone who submitted articles or refereed them for this special issue. We had to omit many good articles because of page limitations. We thank Bruce Shriver for all his encouragement in turning our ideas for a special issue into a good proposal. We also thank Jon Butler for his constant guidance and direction in developing and organizing the issue. Adeline Tang handled the bulk of the correspondence on behalf of the guest editors. We thank her and Carla Shaw, Butler's assistant, who handled all our communications very efficiently.

A. Desai Narasimhalu is a program manager of the Institute of Systems Science, an applied software research organization with the National University of Singapore. His interests include multimedia information systems, user interfaces, object-oriented and heterogeneous databases, and expert systems. He has been the organizing chair, the program chair, and a program committee member for many international conferences and workshops. He is on the editorial committees of *Information Processing and Management* and *Expert Systems with Applications*. He is a consultant to the National Computer Board of Singapore in the area of multimedia technologies.

Narasimhalu received his BS in 1972 from Madras University, MTech in 1974 from the Indian Institute of Technology, Madras, and PhD in 1983 from the University of Hawaii. He is a member of the IEEE Computer Society.



Stavros Christodoulakis is a professor and chair of the Department of Electronic and Computer Engineering, Technical University of Crete, and director of the Laboratory of Distributed Information Systems and Applications. He is also director of the Multimedia Systems Institute of Crete. He is on leave of absence from the Department of Computer Science, University of Waterloo, Canada, where he established and directed the Multimedia Lab. He has been a consultant to industry, and he has authored many research articles in the areas of databases, information systems, multimedia, and optical disks. He is the program committee chair for the VLDB '92 conference on very large databases.

Christodoulakis has a BS in physics from the National University of Greece, Athens, Greece; an MSc in computer information sciences from Queen's University, Kingston, Ontario; and a PhD in computer science from the Department of Computer Science, University of Toronto.



Readers can contact Narasimhalu at the Institute of Systems Science, National University of Singapore, Heng Mui Keng Terrace, Singapore 0511, e-mail issad@nusvm, fax +65-778-2571.

INVALUABLE TO PROGRAMMERS WORKING WITH IBM SYSTEMS!

VSAM
A Comprehensive Guide
By Constantine Kaniklidis



Here is a complete and practical guide to the structure, programming, and optimization of VSAM. Comprehensive and technically sophisticated, this guide includes extensive coverage of the new VSAM (DFP/VSAM), and of VSAM performance tuning techniques (detailed procedures and guidelines). It also explains the intricacies of Linear Data Sets (newest cluster), VSAM Data Set Sharing, Local Shared Resources under CICS, VSAM Hiperspaces, CICS Data Tables, Buffer Management and Optimization under both MVS/XA and ESA, and VS COBOL II's "File Status 2."

\$29.95, 440 pages, 114 line drawings, 0-442-24641-2

To Order: Call Toll-Free 1-800-926-2665 (credit card orders only)



or write to:

VAN NOSTRAND REINHOLD

Mail Order Dept., P.O. Box 668, Florence, KY 41022-0668

K1266

10/91

Advances in Interactive Digital Multimedia Systems

Edward A. Fox, Virginia Polytechnic Institute and State University

*also CACM April '91
Fox's intro*

Progress in hardware, algorithms, and standards has facilitated interactive access to digital multimedia information. This tutorial introduces basic concepts in digital multimedia systems and surveys recent literature.

Humans communicate using a variety of senses and capabilities, especially in face-to-face situations. We should aim to emulate the bandwidth, fidelity, and effectiveness possible in those situations when we develop interactive multimedia computing systems, especially as we move from analog to digital processing environments.

That movement, a part of the evolution of information technology since the early days of computing, gained momentum with the widespread use of compact discs, which demonstrated the accurate reproduction and superb quality of digital audio. Bilevel (black-and-white) image handling, especially facsimile, has demonstrated the potential for rapid communication of documents, changing in a few years the way organizations operate.

Methods for managing computer graphics, color images, and motion video will lead to even greater changes. When fully digital multimedia computing systems are readily available, we will have powerful tools for improving human-human collaboration and human-computer symbiosis.

Televisions, CD players, telephones, and home computers will evolve and be combined, yielding systems with stereo speakers, high-resolution color displays, megabytes of RAM, fast processors for video and audio, fiber-optic network connections, hundreds of megabytes of disk capacity, CD-ROM drives, and flexible input devices, including stereo microphones, pointing devices, and text-entry units. True programming of video will be possible for personalized presentations.

High-resolution images, high-fidelity audio, nicely typeset text, and high-definition video will be available on demand, as versatile alternatives to conventional photographic, audio, newspaper, and television services. Home shopping, cottage industries, delivery of professional services, supplemental adult and child education, surrogate travel to real or artificial sites, video mail and conferencing, and diverse modes of entertainment will be supported.

Many areas of computer science and electrical engineering are aiding these developments. Fast processors, high-speed networks, large-capacity storage devices, new algorithms and data structures, graphics systems, innovative methods for human-computer interaction, real-time operating systems, object-oriented programming, information storage and retrieval, hypertext and hypermedia, languages for scripting, parallel processing methods, and complex architectures for distributed systems — all are involved.

To understand interactive digital multimedia computing systems, it is necessary to see how relevant aspects of these fields relate. The references cited in this article are a small assemblage of quality writings covering much of this broad spectrum.*

First, I provide background regarding developments in interactive videodiscs, which first made images and video accessible through computer systems. Then, I deal with digital storage media, including optical, magnetic, and network options, which allow digital multimedia to be preserved, shared, and distributed. I also discuss the characteristics of audio and video and their digital representations. Because these media are so demanding of space and channel bandwidth, I review compression methods.

Technology is not all that is necessary. Vendors must follow standards — discussed in their own section — to ensure that the economics and usability of digital multimedia help this industry grow. Building on existing de facto standards and an emerging suite of international standards, digital multimedia systems (for example, from Intel, Commodore, and Philips) are already available, and the future of digital multimedia in general looks bright.

Interactive videodiscs

Computer handling of large quantities of audio and video information became possible with the advent of the videodisc in the late 1970s. Each side of these optical discs can hold 54,000 im-

ages, or 30 minutes of motion video if the images are played in sequence at the standard rate of 30 frames per second and they run concurrently with 30 minutes of stereo sound, all recorded in an analog format. Although seek time is on the order of a second, the random-access capability allows computers to control playback in interactive videodisc systems. Videodisc output usually goes directly to a monitor; with additional boards the computer system can overlay text or graphics on the video output, or even digitize the video signal as it is received.

Preparation of videodisc applications is typically a relatively expensive process, requiring a team for design, video and audio production, graphic art, programming, project management, and content specialist duties.¹ While mastering and replication cost several thousand dollars, complete projects may cost \$100,000 per disc. Recordable videodiscs are available but not common, so preparing videodiscs is essentially a publication process. When interactive videodiscs are coupled with high-quality software and a good user interface, powerful educational experiences for thousands of people can result.²

Videodisc applications presenting neuroanatomy and supporting surrogate or simulated travel, language study, and video research work have been developed at MIT in connection with Project Athena and the Media Lab,³ thanks to an elaborate cable plant for analog video. Electronic books, manuals, magazines, and visual databases also have been prototyped there, showing the potential of combining interactive computer systems with videodiscs, graphics, and digitizer boards. However, similar cable systems are too expensive for widespread deployment. Instead, the shift is to digital storage and communication systems.

Digital storage media

In the jargon of the international standards community, "digital storage media" refers to both storage and communication capabilities. These have increased in capacity more than a thousandfold in the past decade, allowing digital multimedia to emerge as a cost-effective competitor for analog approaches. In 1985, CD-ROM (compact-disc read-only memory) with a capacity

of more than 600 Mbytes but a manufactured cost under a dollar was first applied to electronic publishing.^{4,5} Similarly, fiber-optic cable now makes gigabit-per-second communication channels a reality worldwide.

Storage. CD-ROMs, the most cost-effective storage medium for distributing large quantities of digital data, are related to regular compact discs — that is, CD-DA, or compact-disc digital-audio. Both support direct access to individual sectors of data that can store 1/75 second of CD-quality digital audio in CD-DA format, using 2,336 bytes, or 2,048 bytes of arbitrary digital data in CD-ROM format, with the rest of the space for error correction. Thus, CD-ROM data transfer speed is 150 Kbytes per second or 1.2 Mbits per second.

An elaborate system of optics, servos, and signal decoding circuits allows data to be accessed within a second.⁶ International Organization for Standardization (ISO) 9660 (based on the High Sierra standard⁷) specifies the volume and file characteristics, allowing access through nearly any CD-ROM drive and operating system. Efforts of the Rock Ridge Group have extended the utility of ISO 9660 for Unix systems and servers.

Compact discs are part of the family of optical media.^{8,9} There are prospects for further improvement, including write-once and erasable discs of varying sizes.¹⁰ For example, announcements indicate that in 1992, write-once CD-ROM drives will cost under \$3,000, and minidisk (2.5-inch, 128-Mbyte) drives will cost about the same as those for CD-ROMs, handling both formats. Further information on optical publishing and access is available in the literature.^{4,5}

Magnetic disks are readily available with capacities on the order of a gigabyte, but they are not low-cost, removable, or produced by mass replication. Use of caching, memory hierarchies, and minimal perfect hashing will make all storage units even more effective contributors to digital multimedia.

Networking. While fiber-optic connections are the most cost-effective scheme for rapidly transmitting large volumes of digital data, a great deal of research regarding networked multimedia is required. The Fiber Distributed Data Interface operates at 100 Mbps,

* This tutorial provides an introduction to the basic concepts and an overview of the literature; a longer version is available from the author, whose address appears after his biography on p. 21.

Table 1. Multimedia publishing.

Medium	Collecting	Preparing
Video	Video edit list	Digitize, enhance, compress
Image	Selection	Digitize, enhance, compress
Audio	Audio edit list	Filter, digitize, enhance, compress
Text	Text filing	Text processing, tool use
Graphics	(Composite) Object selection	Rendering, drawing, animating tool use

Publishing: ordering, correlating multiple representations, organizing (layout, order), hypermedia linking, adding interactive scripts.

Using: network distribution, optical publishing, on-line use, hypermedia enhancement by users.

and connection costs dropped below \$5,000 in the early 1990s. Gigabit-per-second networks are being tested, and will form the backbone for national telecommunication. However, research regarding network protocols, local area network and workstation architectures, and operating-system support software for digital multimedia is still in its infancy. The first international workshop exclusively on these topics was held late in 1990.¹¹

Asynchronous transfer mode networks support both variable and constant bit rate services at very high speeds and serve as the basis for the Broadband Integrated Services Digital Network.¹² Streams of digital multimedia, including uncompressed live video, can use the constant bit rate services. These types of fast packet switching can support real-time communication with reduced jitter. (Jitter is caused by delays of packets.) Yet many questions remain regarding synchronization of related data types, dynamic adaptation to different terminal and network services, and real-time requirements for hypermedia.¹³ Research is also needed on rate structures, spectrum and channel allocation, and handling of scalable representations of video streams.¹⁴

Network and operating-system concerns coalesce because of the need for open systems, synchronization ("lip-sync" of audio and video), and fast transfer from network to presentation unit. High performance and reliability are also important. Research is proceeding on object-oriented models to handle the

various classes of multimedia,¹⁵ abstractions for continuous media (for example, an audio stream) I/O,¹⁶ and connection architectures for networks and workstations.^{17,18} There is a trade-off between network reliability, how closely the network approximates providing a constant rate stream, and other factors. Without buffering, the stream of data must be carefully prepared, such as by interleaving data on storage units.¹⁹ That is especially important with CD-ROM, where seek times are one to two orders of magnitude slower than with magnetic disks. As memory prices decline, however, buffering of multimedia data will allow fast processors to compensate for many performance problems in digital storage media.

Adding audio and video capabilities

While issues of text and graphics processing have continuing importance in standard information systems, the current wave of development in digital multimedia deals mainly with audio and video. Luther²⁰ discusses many important issues, and Bottoms and Helgeson²¹ and Adkins²² cover ways to manage data-conversion projects including these media. The goal is integrated capabilities encompassing cameras, slide and filmstrip projectors, camcorders and VCRs, tape recorders, television broadcast and reception units, and postproduction studios.

Many tools are required to fully support both commercial and end-user multimedia publishing. Table 1 illustrates some key issues for important media types. First, suitable multimedia objects must be collected. These come from natural or artificial sources—for example, a captured image or a synthesized sound. While many people know how to file text documents, select a graphic object while drawing, or choose slides for a presentation, most people have had no experience editing audio or video files. Nevertheless, tens or hundreds of hours of raw footage are often edited to produce a single hour of a video production.

After collection comes preparation. For use with computers, audio and video are digitized and then stored in the minimum amount of space suitable for subsequent use. Specialized tools for each medium are required to enhance or add new materials. Multimedia publishing then involves ordering the pool of accumulated resources and tying together the various representations of each object (for example, audio and video synchronization or sequencing of images obtained at different times). Publications must be organized both spatially and temporally, and associational links must be provided, as with hypermedia.²³ Suitable sequences of interaction must be scripted.²⁴ The resulting publication is shared over a network, distributed on CD-ROM, or enhanced into an even better publication as part of a hyperbase (a database of hypermedia documents).

Multimedia publications stretch the capabilities of modern computers. Enormous requirements for storage make compression necessary. Computers must provide facilities for accurate control of playback, flexible editing, and signal processing or enhancement. Combinations such as warping video onto graphic structures must be supported.

Audio. On a computer, digital sounds can supplement video, communicate when the user's other senses are engaged, or provide a suitable background or stimulus. Brewer discusses uses of audio as well as technical issues relating to CD-ROM.²⁴ While special VLSI chips have been developed for various digital signal processing tasks, the related matter of digital representation is a particular concern.

When sound is digitized, it is usually

PCM
sampled and quantized using a scheme called pulse code modulation. According to sampling theory, samples should be taken at least twice for each cycle of the highest frequency component to be recovered. Because the human ear is not sensitive to sounds higher than 20 kHz, the 44.1-kHz sampling rate used for CD-DA permits fairly accurate reproduction. Each sample uses 16 bits to indicate the amplitude — that is, to quantize the sound — yielding a large dynamic range and a signal-to-noise ratio of over 95 dB. The main disadvantage of this approach is that roughly 10 Mbytes of storage are consumed by each minute of audio (171 Kbytes per second). In other words, a communication channel of about 1.4 Mbps must be allocated to handle a single stereo source.

Several approaches are commonly used to reduce the need for storage, but they reduce the quality of reproduction. Storage decreases when a system uses

- mono instead of stereo;
- fewer samples, reducing the frequency range covered; or
- fewer bits per sample, reducing the quality of waveform reconstruction.

Thus, with Compact Disc-Interactive, or CD-I, level A stereo sound requires 85 Kbytes per second, level B 42.5 Kbytes per second, and level C 21.3 Kbytes per second to give sound quality comparable to an LP record, FM radio, or AM radio, respectively. Significant further reductions are possible with adaptive differential pulse code modulation schemes, which use fewer bits for quantization. The differential approach involves recording the difference from the previous sample and often requires fewer bits than the actual current value. The adaptive feature involves computing parameters, so the scale for values changes as the rate of amplitude shift varies. Often, 4 bits per sample is adequate for good-quality reproduction, as in the ADPCM scheme used in the CD-ROM XA (extended architecture) standard promulgated by Philips and Sony.

Graphics, images, and video. Doherty and Apperson survey graphics, images, and video.²⁵ Many issues relate to all forms of graphics, images, and video, but there are differences in origin, storage, and presentation. Generally, graphics components of multimedia programs originate on computer systems. It is

Most current representations of images assume that descriptions are a two-dimensional array or raster, and that video is a sequence of such images.

possible for natural images to lead to graphics, such as when line-tracking algorithms are applied to digitized versions of engineering diagrams, or when scanned images are converted to a "draw" representation by software that tries to fit lines and filled shapes. Sophisticated conversions to the structured representation characteristic of graphics or to higher level forms managed by artificial intelligence routines also take place in the context of computer vision.

Ultimately, images and video will be analyzed and stored in high-level and storage-efficient representations that identify and characterize objects, relationships, distances, and movement. This will facilitate "virtual reality" investigations, support model-based compression, and allow truly scalable presentations for displays varying in size from wristwatch monitors to wall panels.²⁶

Most current representations of images assume that descriptions are a two-dimensional array or raster, and that video is a sequence of such images. The "resolution" of an image is crucial as it defines the raster size. Still images that are closely examined should be stored using at least 512 × 480-pixel resolution, according to experiments we have conducted at Virginia Polytechnic Institute and State University; but, in such applications as menu selection or image database browsing, low-resolution picture icons or "picons" may suffice.

With video, the key issue is the enormous bandwidth and storage required for digital representations. NTSC television (the US standard established by the National Television System Committee) has 525 horizontal lines to define the vertical resolution or height. The theoretical width or horizontal resolution limit, based on the allowed bandwidth of 4.5 MHz, is 360 vertical lines. However, because of the scanning meth-

od and television construction constraints, the "safe" region on a television is even smaller, so 360 × 240 pixels is often the resolution achieved. This is called "normal resolution" for CD-I.²⁷

Recommendation 601 of the International Radio Consultative Committee, or CCIR, specifies the international standard for digital video as 720 lines. The number of pixels per line varies depending on the television scheme used: There are 480 pixels for NTSC and 576 for PAL²⁸ (phase alternating line, the European standard). The International Telegraphy and Telephony Consultative Committee, or CCITT, uses the Common Intermediate Format (CIF) for video telephony, which has a resolution of 360 × 288 pixels, and Quarter-CIF (QCIF), which has a 180 × 144-pixel resolution.²⁹ Super VHS camcorders achieve 400 lines, and high-definition television about 1,000 lines.

Images have a third dimension, pixel depth, which refers to the number of bits used for each picture element. While monochrome uses a single bit, gray scale often uses 8 bits to get 256 levels. For color graphics, 4 or 8 bits may suffice, while for color images, 8, 9, 16, or 24 bits are standard. With 32-bit color, 24 bits are used for color and 8 bits for the "alpha channel," which indicates degree of transparency or mixing with other image planes.

Pixel depth relates to the choice of color space. Cameras or scanners will often separate the red, green, and blue (RGB) components and quantize them with a certain number of bits each. Examples are RGB 5:5:5, which is used with CD-I,³⁰ or 8:8:8, which produces 24-bit color. However, the human visual system is not as sensitive to color (chrominance) as it is to intensity (luminance), so color spaces other than RGB have been developed. Mappings between these color spaces can be done with linear transformations.³¹ For PAL, the YUV space was chosen, and that has been adopted for use with both CD-I and Intel's Digital Video Interactive, or DVI. For analog television, 4.5 MHz can be used for luminance (Y), and 1.5 MHz each for the two chrominance channels (U and V).

Home televisions and VCRs, which are considered acceptable by many viewers, often use only 0.5 MHz for the chrominance channels. Similarly, with CCIR Recommendation 601, chrominance is subsampled, yielding a 4:2:2

scheme that provides twice as many samples for luminance. DVI's 9-bit compressed video format uses 4:1 subsampling in each dimension, yielding 1 bit of chrominance for every 8 bits of luminance.²⁰

Smooth motion video requires at least 25 to 30 frames per second. PAL and Secam (Sequential Couleur avec Memoire, the standard used in France) use 25 frames per second; NTSC uses 30 frames per second. These are usually interlaced, so each frame is made of two fields containing alternating lines, and two fields are shown during each frame time. Eventually, noninterlaced displays like computer monitors could be refreshed with a full image 60 times per second.

To estimate bandwidth requirements, however, we can concentrate on total samples per second for various schemes. If digital video is uniformly quantized at 8 bits per sample, according to CCIR Recommendation 601, with a sampling rate of 13.5 MHz for luminance, the total bit rate is 216 Mbps. The bit rate for NTSC "network-quality" video is about 45 Mbps.²⁸ For CIF and QCIF the uncompressed bit rates are approximately 36 and 9 Mbps.²⁹ Contrast these rates with the 150-Kbps data rate of CD-ROM!

Compression methods

Clearly, compression is essential if audio, images, and video are to be used in digital multimedia applications. A megabyte of space would be filled by roughly six seconds of CD-quality audio, a single 640 × 480-pixel color image stored using 24 bits per pixel, or a single frame — 1/30 second — of CIF video. Nevertheless, videodisc applications often have more than 20 minutes of video, perhaps 10,000 slides, and 30 minutes of stereo sound on each laser disc side. And with the tremendous volume of data that will be received each day from planned NASA missions³⁰ and other scientific ventures, the need for proven compression techniques is obvious.

Happily, there has been a great deal of research and many implementations using software, hardware, or both for a variety of compression methods. Research continues, with further improvements expected using wavelet and other time- and space-domain schemes. Elsewhere, I have published a brief intro-

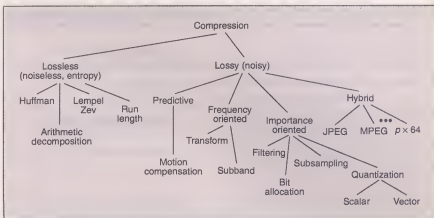


Figure 1. Selected compression approaches.

duction with a glossary and references.³¹ Rao and Yip provide a comprehensive treatment of compression for images and video.³² A set of useful articles appears in conference publications by NASA³³ and the IEEE Computer Society,³⁴ with many good references included in the bibliographies.

Compression of digital data involves computational algorithms that can be implemented in software. Some involve digital versions of signal processing methods, others involve pattern recognition, and still others use statistics or characteristics of particular data types or samples. High-speed implementations involve VLSI chips, such as for audio digital signal processing, discrete cosine transform, or vector quantization approaches. I discuss these in a later section.

Approaches. At the boundary of image processing, computer vision, and graphics is the area of model-based compression.²⁶ Models of faces can be analyzed, yielding facial motion parameters that can be transmitted at low bit rates and synthesized at a receiver for "talking head" video telephony. Other approaches involve feature detection at the encoder and rendering at the decoder.

Fractals (images that can be described by a set of rules specified with a relatively small number of bits) allow compression of natural scenes where the underlying structure matches this type of model.³⁵ Very high compression ratios can be achieved, sometimes on the order of 1,000:1 (size of the uncompressed form versus the compressed form). How-

ever, extensive computation is required for encoding. While decoded images may be acceptable to human judgment, there is usually some quality loss. Nevertheless, several companies — for example, Barnsley Communications — are marketing boards and software for fractal compression and decompression. Commodore has announced plans for software-based fractal decompression in their CDTV system.

Figure 1 shows a taxonomy of compression approaches. In lossless schemes, the original representation can be perfectly recovered. For text, lossless methods may achieve a 2:1 reduction. For bilevel images, 15:1 is a good figure. (A new international standard for bilevel image coding, referred to as JBIG, improves on CCITT Group 3 and 4 approaches for facsimile transmission, and in some situations achieves more than 50:1 compression.)

These approaches are also called *noiseless* — because they do not add noise to the signal — or *entropy coding* — because they eliminate redundancy through statistical or decomposition techniques. For example, Huffman coding uses fewer bits for more common message symbols, and run-length encoding replaces strings of the same symbol with a count/symbol pair. Discussions of research on lossless compression appear elsewhere.^{36,37}

The other approach, *lossy* compression, involves encoding into a form that takes up a relatively small amount of space, but which can be decoded to yield a representation that humans find similar to the original.

Table 2. Standards for multimedia and hypermedia.

Short Name	Official Name	Standards Group	Group Designation	Approval Status
JPEG	Digital compression and coding of continuous-tone still images	Joint Photographic Experts Group	JTC1/SC2/WG10	Committee draft balloting in 1991
H.261	Video coder/decoder for audiovisual services at $p \times 64$ Kbps	Specialist Group on Coding for Visual Telephony	CCITT SG XV	Approved Dec. 1990
MPEG	Coding of moving pictures and associated audio	Moving Picture Experts Group	JTC1/SC2/WG11	Video committee draft balloting in 1991
MHEG	Coded representation of multimedia and hypermedia information	Multimedia and Hypermedia Information Coding Expert Group	JTC1/SC2/WG12	Working document
HyTime	Hypermedia/Time-Based Structuring Language	Standard Music Representation Work Group	ANSI X3V1.8M	Committee draft balloting in 1991

Lossy compression. Lossy or noisy compression may add artifacts that can be perceived.³⁵ Careful study of the human visual system has focused attention on approaches that cause little perceived loss in quality, but achieve high compression ratios.

Prediction. Predictive approaches like ADPCM involve predicting subsequent values by observing previous ones; and transmitting only the usually small differences between actual and predicted data. An example involves motion compensation. Successive frames in a video sequence are often quite similar or have blocks of pixels shifted from one frame to the next — for example, as the camera pans or a person moves.²⁰ Although it is computationally expensive to analyze images and yield motion vectors, parallel computers³⁶ or neural networks can help with the processing.

Frequency-oriented compression. Subband coding can exploit the fact that humans have different sensitivities to various spatial and temporal frequency combinations.³⁷ The idea is to separate (for example, using a series of filters) the different frequency combinations, and then to code with greater fidelity the frequencies that humans pay particular attention to. Without subband coding, all frequency combinations would be coded identically, so the technique achieves high perceived quality with fewer total bits.

Another approach relating to humans'

handling of frequency is transform coding. This usually involves spatial frequencies, as in single images. The most common approach applies the *discrete cosine transform*, which is related to the fast Fourier transform. Rao and Yip cover DCT approaches thoroughly.³² Lower spatial frequencies must be carefully coded, while higher frequencies need less detailed coding. If we think of a block (say, an 8×8 -pixel section) of a two-dimensional image as a square with rows and columns numbered from the top-left corner, then the DCT of that block will also be a similarly numbered square. Consider a zigzag sequencing of the values in transform space, starting at the top-left corner and covering the nearest cells first. Run-length encoding and coarse quantification of cells later in the sequence both lead to good compression. The encoder applies DCT in the forward direction, and the decoder uses an inverse mapping from transform to image space.

Importance-oriented compression. Other characteristics of images besides frequency are used as the basis for compression. The principle is to consider as more important those parts of an image that humans are better attuned to. An example of this approach is to filter images, getting rid of details that cannot be perceived, as in the low-pass filtering done for real-time video with DVI systems. Another technique is to allocate more bits to encode important parts of an image, such as where edges occur,

than to encode large homogeneous regions, such as those depicting clouds.

Color lookup table use. as in CD-I and DVI, applies the principle of indirection. Instead of letting the bits that describe a pixel refer to a location in color space, the bits identify a table location, and the table entry refers to color space. Color spaces often cover a palette of size 2^8 , which means 24 bits are needed. On the other hand, lookup table size may be only $256 (2^8)$. The reduction is 24 to 8 bits per pixel. The challenge is to select for each lookup table the most important colors to be accessed by the display processor.

Subsampling, also based on characteristics of human vision, was discussed in an earlier section. It involves using fewer bits for chrominance than luminance. Interpolation, which can be carried out in hardware,⁴⁰ results in a full but approximate reconstruction of the original. We can think of this process as that of taking one matrix and generating from it another matrix four or 16 times larger — by interpolating values horizontally, vertically, and diagonally. Related to interpolation is line doubling, used in some DVI systems to go from the 256 lines that result from video decompression to 512 lines.

Importance also relates to patterns in an image representation. Clearly, higher level descriptions where symbols refer to large structures can take much less space than raster forms. In coding theory, this translates into the fact that vector quantization can lead to higher

compression than scalar quantization.³⁹ Scalar quantization is often just called quantization, and was discussed in an earlier section in connection with pulse code modulation and audio encoding. It takes values and maps them into a fixed number of bits.

Vector quantization, on the other hand, usually takes two-dimensional vectors of values — for example, 4×4 — and maps them into a code symbol. Thus, code books are developed for images, recording the most important vectors, and all data vectors are mapped to the nearest code-book entry, minimizing mean square error. Decoding involves fast table lookup to replace coded entries with vectors from the code book.

Vector quantization is discussed in depth in the literature.^{41,42} There are many algorithms and some VLSI implementations. Encoding usually takes a good deal of computation, so near-optimal code books can be developed.

Hybrid coding. Various compression approaches can be combined, for example, DCT and differential pulse code modulation, subband coding and DCT, or differential pulse code modulation and vector quantization. Generally, subband coding is coupled with vector quantization. Systems and standards for video compression often apply motion compensation for temporal compression, transform coding for spatial compression, and Huffman or arithmetic coding for statistical compression.

Standards

Initially, standards for digital multimedia were established by fiat. Thus, Philips and Sony developed CD-I, with its formats for images and audio.⁸ Similarly, as a step toward CD-I and a way to incorporate audio more easily into CD-ROM applications, the companies specified CD-ROM XA. It uses ADPCM, with three different sample rates (levels A, B, C). Since the specification of CD-ROM XA, manufacturers have produced interface boards for CD-ROM-based playback. Very widespread use of CD-ROM XA is expected, such as through joint efforts involving Philips and Nintendo systems.

Recently, important new international standards for images, audio, and video have been accepted by electronics, computer, and communications organiza-

Important new international standards for images, audio, and video have been accepted through consensus.

tions through consensus, with the standards groups actually pushing the limits of state-of-the-art research. Table 2 presents summary details; explanations follow in subsequent sections. Two types of standards are particularly important: low-level coding or compression standards for data streams and hardware processing, and higher level standards for network and software operation.

Coding. The first standard listed in Table 2 is JPEG,⁴³ named for the Joint Photographic Experts Group, which developed it. This work is technically complete. The committee draft was sent for balloting early in 1991, and it will be approved as ISO/IEC (International Electrotechnical Commission) Standard 10918 by 1992. Processing usually involves sequential encoding using forward DCT scalar quantization, and either Huffman or arithmetic coding for compression. The decoder simply reverses the process. Applications control the quantization and Huffman tables.

JPEG also includes optional modes of compression and decompression. Progressive encoding is supported, where the image is encoded in multiple scans and the viewer can see the image build up in successively more detailed versions. This operation is important when communication bandwidth is low, or users need to browse through an image collection. Hierarchical encoding is also specified, where lower resolution images can be accessed before higher resolution images, again useful for browsing or low-resolution displays. JPEG also specifies a lossless encoding scheme, which involves a predictor and entropy coder.

A second international standard, CCITT Recommendation H.261, is referred to as $p \times 64$.²⁹ The p is a parameter since the standard can be used to pro-

duce compressed video streams at rates of $p \times 64$ Kbps, with p ranging from 1 to 30. Given CIF or QCIF input, $p \times 64$ codecs (coder/decoders) will support video telephony. For $p = 1$ or 2, only QCIF and videophone applications at low frame rates will be possible, but for p over 5, use of CIF and video conferencing is possible.

The coding is a hybrid of DCT using 8×8 blocks, and differential pulse code modulation with motion estimation²⁹ involving 16×16 luminance blocks, using picture memory for macroblock comparisons. Spatial coding involves DCT followed by scalar quantization. The loop filter removes high-frequency noise. Depending on the fullness of the output buffer, the quantizer step size can be varied to decrease or increase the bit rate. Statistical coding takes place in the multiplexer.

A third important standard is referred to as MPEG, for the Motion Picture Experts Group.⁴⁴ The ISO/IEC committee draft was assigned standard number 11172. There are actually three parts to the effort: MPEG-Video, MPEG-Audio, and MPEG-System. The first refers to work with video compression of television resolution — 360×240 pixels — down to a bit rate of approximately 1.2 Mbps. This rate is suitable for use with CD-ROM, digital audio tape, and T1 communication channels. All MPEG decoders should be able to operate with a "core" bit stream that has the following upper bounds⁴⁴: resolution of 720×576 pixels, speed of 30 frames per second, bit rate of 1.86 Mbps, and decoder buffer of 46 Kbytes. Another MPEG project is to investigate compression for CCIR Recommendation 601 video at bit rates up to 10 Mbps.

While there is overlap between $p \times 64$ and MPEG capabilities, and some of the VLSI components involved can be used for both, $p \times 64$ has a goal of broad coverage for varying channel capacity, while MPEG has the goal of high-quality coverage at an important but narrower range of bit rates.

MPEG-Video specifies a layered bit stream that consists of a video sequence header and layers for sequence, a group of pictures, pictures, slices, macroblocks, and blocks.⁴⁴ The lower two levels include motion compensation and DCT data. The algorithm ensures features desirable for a range of applications: random access, fast forward and reverse searches, forward and reverse playback,

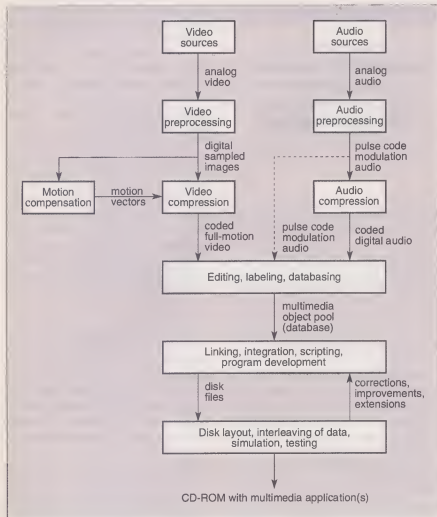


Figure 2. MPEG-based CD-ROM development.

audio/video synchronization, robustness against errors, delays limited to 150 ms, and editability.

The MPEG-Video algorithm uses motion compensation on 16×16 blocks and DCT coding of 8×8 blocks, followed by quantization and entropy coding. Motion compensation involves predictive and interpolative coding. As in JPEG, reference frames are fully coded using DCT-based methods. Predicted (interpolated) frames are determined on the basis of the nearest reference or predicted frame (on both sides). Finally, the motion information is statistically coded.

The MPEG-Audio standard⁴⁵ is being fully specified in 1991. While CD-quality audio requires two channels of 706 Kbps, the compressed bit rate will be two channels of 128 kbps or possibly

64 Kbps — that is, a reduction from 1.412 Mbps to 0.256 Mbps. Input sampling rates of 32, 44.1, and 48 kHz with 16 bits per sample will be supported; delays will be no more than 80 ms; and addressing will be to units no longer than 1/30 second. Sound quality close to the original is expected.

Ultimately, MPEG-System work will produce a complete approach to encoding television-quality audio and video into a single stream operating at about 1.5 Mbps. Figure 2 shows how this may lead to development of digital multimedia applications for CD-ROM. Video will be converted to digital form and then compressed using motion compensation and other processing specified in MPEG-Video. Audio will be digitized. While limited amounts of pulse code modulation audio can be stored, most

will be coded according to the MPEG-Audio standard. Application development will then proceed, involving editing, labeling, and cataloging of the audio/video resources into a pool of "objects." Higher level activities of linking, integrating media streams, scripting interaction, and developing programs all lead to stored files. For CD-ROM, disc layout of possibly interleaved data allows simulation to ensure proper real-time operation of finished applications. After enough cycles of testing and refinement, a finished CD-ROM master can be used for replication.

With these several standards, it is imperative that headers and descriptors for video streams be created so coding methods can be easily distinguished.¹⁴ This is particularly important for multimedia electronic mail.⁴⁶ These standards will certainly be further developed to better support users of digital multimedia.^{2,5}

High level. In addition to coding, standards are needed for higher layers of the multimedia application development process. This work is just beginning, and it will require a few years to reach maturity. One effort is coordinated by the Multimedia and Hypermedia Information Coding Expert Group, or MHEG, whose aim is to allow bit-stream specifications for multimedia and hypermedia applications on any platform, leading to good real-time performance drawing on object-oriented methods. Version 2, July 1990, of the MHEG Working Document S includes useful definitions regarding multimedia and hypermedia. It gives a partial object hierarchy of both basic and composite objects, covering input, output, interaction, and linking. Methods include editing, operating, and presenting, with a state model including combinations of not ready versus stopped/ready, running versus suspended, and exposed versus concealed. MHEG covers synchronization (between objects, or using marks or conditions), buffer memory, input objects like buttons or menus, and interactive objects like prompts.

A higher level but related effort involves extending ODA (Office Document Architecture) standards to include hypermedia. Further along, however, is work in applying the Standard Generalized Markup Language (SGML, ISO 8879) to multimedia and hypermedia. This approach led to HyTime,⁴⁷ ballot-

ed in 1991 as ISO/IEC Committee Draft 10744. HyTime allows a complete digital multimedia application to be encoded in a linear stream, including all details regarding "document" structure, hypermedia linking, synchronization, and timing. For interchange, Abstract Syntax Notation 1 (ASN.1, ISO 8824) allows bit string representation. A pool of multimedia objects can be referenced from the main part of the document. Also, there are series of events and "batons" to control the timing, with mapping from virtual time units to real time. The Text Encoding Initiative has drawn on HyTime for describing multimedia and hypermedia documents and performances.

Further standards work will facilitate interchange and transcoding of multimedia,¹⁴ integration of object pools with scripting languages, interoperability of applications on heterogeneous platforms connected by fiber networks, conversions between storage units, and other requirements called for by a growing community of users.

Systems

Interactive multimedia computing systems must support application development and use. At present, there is a tight coupling of hardware, software, and applications, but that coupling will gradually loosen. Here, I deal with a variety of platforms and related software.

Next. Next systems with a Nextdimension board support 32-bit color (24-bit color plus an alpha channel). The board was designed to include an i860 processor for graphics and a JPEG compression chip¹⁸ from C-Cube Microsystems. While 640 x 480-pixel image compression in software requires almost 10 seconds using the Motorola MC68040 microprocessor, the special board must do that in 1/30 second. This allows full-motion compression and decompression, using the 600-Kbps speed achieved by some hard disks to achieve moderate-quality playback.

MediaView is a digital multimedia communication software system built with Nextstep, the interface used on Next systems, to handle graphics, audio, video, animations, mathematics, and so on.⁴⁹ It uses an object-oriented approach to documents. Searching is sup-

Interactive multimedia computing systems must support application development and use.

ported for text, and digital darkroom methods are used for images. Annotations include Draw-it and Hear-it notes, as well as object- or image-based animations. Full-motion video will be supported through the Nextdimension board.

CD-I, CDTV. Compact Disc Interactive was the first interactive digital multimedia technology targeted for the consumer market. First announced in 1986, CD-I is defined in a proprietary standard, the "Green Book," which Philips provided to developers under a license arrangement.⁹ The first prototype disks were prepared in 1987. The official commercial launch of CD-I was delayed until fall of 1991, reflecting the complexity of software and application development. Each unit includes a compact disc player, a Motorola M68000 family microprocessor, an audio processing unit, audio and video decoders, a clock, a pointing device, and an operating system, CD-RTOS, derived from OS-9, with real-time capabilities. There will be optional MPEG-compliant full-motion video support. The minimal system, the CD-I Base Case Decoder, will connect to an external monitor and external speakers. The decoder supports a variety of visual effects, including single-plane effects like cuts, scrolling, mosaics, and fades, as well as two-plane effects like transparency, chroma key, matte, dissolve, and wipe.

The challenge of CD-I has been to build a carefully defined hardware, software, and storage environment for interactive multimedia computing that can be sold as a package for under \$1,000. Such systems would serve the needs of consumers and the low-end education/training marketplace. Careful design is required, since "hit" CD-I discs, selling for perhaps \$40, might establish a large consumer electronics market, as did compact disc players and Nintendo units. Toward that end, Optimage Interactive Services Companies has developed tools

to help developers, and there is Starter System software that works with the standard decoder, allowing previously captured materials to be reorganized for presentations. However, full application development currently calls for PC-, Macintosh-, and/or Sun-based networks and a variety of software tools and programming expertise.

Bruno gives a short introduction to CD-I.²⁷ Brewer describes CD-ROMXA, which specifies the use of various levels of CD-I audio in an open hardware environment.²⁴ Philips' work with MPEG compression and related details on the video decoder are discussed by Sijstermans and van der Meer.²⁸

In the spring of 1991, Commodore announced CDTV (Commodore Dynamic Total Vision), with pricing and an approach similar to those of CD-I. Commodore uses CD-ROM with proprietary representation schemes rather than the CD-I format, so CDTV is positioned as a related but competing technology, building on Commodore's experience with real-time operating systems and the Amiga 500 computer.

DVI. Intel's Digital Video Interactive is a more open approach than CD-I or CDTV, emphasizing digital video and audio compression, chip sets, and machine-independent audio/video software environments. Luther gives an excellent short introduction³⁰ and the best overall description.²⁰ Included in a series of articles in a special section in *Communications of the ACM*³¹ is an overview of the technology and applications, a short history of the development and graphics-based prototyping, a description of parallel encoding, and details of an exciting application to training. Elsewhere, Harney et al. described the new "B" chip set,⁴⁰ and Green described the new software design.⁵²

Current DVI technology uses a 25-MIPS pixel processor, the 82750 PB. Although in some ways it is like a conventional microprocessor, the 82750 PB is five to 10 times faster because of its specialized architecture. It is programmable and can support various standards.⁴⁰ The 82750 PB can

- perform graphics operations rapidly;
- use special effects to highlight image or video transitions;
- decode a 640 x 480-pixel JPEG compressed image in less than a second;

Table 3. Classes of multimedia systems.

Component	Multimedia PC	Typical Current System	Typical Advanced System
Storage	CD-ROM, 30-Mbyte disk	CD-ROM, 100-Mbyte disk	CD-ROM, gigabyte disk
Host processor	80286	80386 or MC68030	80486 or MC68040
Display	VGA	768 × 480	1,024 × 768
Extra board	Audio board	Nextdimension, CD-ROM XA, DVI	MPEG system
Networking	None	Ethernet, token ring	Fiber Distributed Data Interface
Software	Windows 3.0	Authology: Multimedia, Mediascript, MediaView	Athena Muse, Integrator

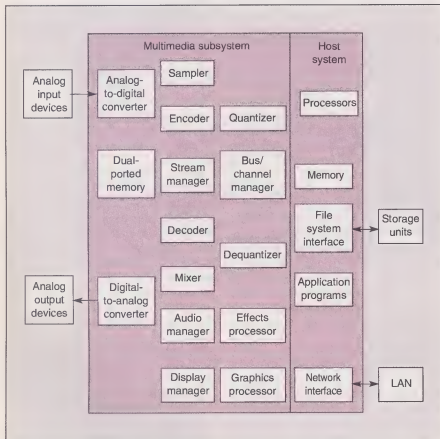


Figure 3. Logical components of a future multimedia workstation.

- compress or decompress images using a proprietary format at up to 768 × 480-pixel resolution in a fraction of a second;
- compress 128 × 120-pixel resolution motion video at 30 frames per second with fair quality into a special representation (RTV, or real-time video) on hard disk; and

- play back video compressed on a 64-processor Intel computer into another special representation (PLV, or production-level video) at 30 frames per second with good quality at 256 × 240-pixel resolution.

Combinations of these operations are fast and effective for such effects as

warping textures selected from an image database onto graphic objects selected from a wireframe database.

Intel has published a bit-stream specification. DVI will support MPEG standards when they are fully specified, and Intel has an even faster pixel processor planned for 1993.

The companion display processor, the 82750 DB chip, supports pixel interpolation, genlocking, an alpha channel, reconfigurable color lookup tables, conversion from YUV to RGB, and triple 8-bit digital-to-analog converters, for resolutions at least through 640 × 480 pixels. Intel provides the ActionMedia 750 board set for AT or Micro Channel Architecture 80386-based computers. Boards using the chip set have been developed for Macintosh computers as well.

The present DVI audio/video subsystem software, called AVSS, will be replaced by AVK, the audio/video kernel.³² Instead of the current single environment for DOS, AVK has been designed to work with at least Microsoft Windows, OS/2, and Unix. Instead of implementing only the simple VCR metaphor supported by AVSS, AVK uses the metaphor of a digital production studio, managing a variety of streams with sampling, mixing, and effects. More attention is given to interfacing with the host operating system and handling real-time operations. Saturation of processors, memory, and I/O is avoided with flexible buffering. Scheduling is handled by the pixel processor instead of the host system, streams are synchronized into groups, and windows are supported.

Third-party developers have provided higher level software support. For

example, Mediascript²⁰ is a high-level language interpreter that eliminates much of the need for programming in C. Other tools support painting (for example, Lumena), animation building (Autology: Multimedia), and hypermedia. Our efforts at Virginia Polytechnic Institute and State University (VPI & SU) have led to a "hot spot" tool for managing buttons and a prototype "integrator" that uses the metaphor of a musical score and concepts covered by the MHEG and HyTime standards as the basis for direct manipulation to develop interactive multimedia applications.

Prospects

Clearly there are opposing forces in the future of digital multimedia systems: closed versus open systems, low cost versus high cost, performance versus flexibility, and stand-alone systems versus distributed systems. Table 3 illustrates two key dimensions of systems: components and power. Companies like Microsoft and Tandy have promulgated multimedia PCs (MPCs), but these are fairly minimal systems. As the table shows, most serious offerings go beyond these minimal specifications, possibly adding Ethernet or token ring networking.

The next generation of digital multimedia systems will be at the level shown in the last column of Table 3. Not only CD-ROM but also large-capacity writable local storage will allow capture and development as well as use of applications. Faster host processors and full support for the MPEG standard will be required. Networking will provide access to image, audio, and video servers at high speeds. Tools like our integrator or systems building on the Athena Muse system at MIT⁷ will provide flexible support for a variety of application classes.

Figure 3 illustrates one type of future workstation. Access to network and storage units as well as analog input and output devices must be supported. Inside the workstation (the workstation itself may be physically distributed) will be the standard host system components and a comprehensive multimedia system. The multimedia system may be part of the host motherboard, but we can think of it as logically separate. Hardware and software components must carry out the multiple functions shown for the multimedia subsystem.

Table 4: Reference model, upper layers.

Layer	Description/Contents
Application Presentation	Hypertext, hypermedia, interactive multimedia Devices: keyboard, mouse, speakers, windows Media: animations, audio, graphics, images, video Operations: accessing, editing, sequencing, transforming
Session Base	History management, user identification, versioning Searching and inferring with data, information, and knowledge bases
View	Frames, graphs, lists, relations
Link	Labels, link identifiers, types
Anchor	Anchor identifiers, span descriptions
Node	Basic and composite media objects, object identifiers
Communication	File systems, messages, processes

Table 4 illustrates how these advanced workstations might relate to current views of interoperability. Above the bottom layer shown here, which represents the lower levels of the Open Systems Interconnection reference model, are four layers related to multimedia and hypermedia, followed by the usual three top layers of the OSI reference model. Thus, the view layer provides convenient abstractions that support data, information, and knowledge bases, which fit into the next layer. In our work at VPI&SU on the Large External Network (object-oriented) Database, we have implemented parts of these top five layers. Altogether, Table 4 provides a partial conceptual framework for digital hypermedia systems, emphasizing how they must fit into the overall picture of open networks, information processing, and flexible access.

and tools will mature, and publishers and end-users will learn to work creatively with a range of new capabilities.²³ Interactive computing will benefit from the development of the audio and television industries, leading to exciting new possibilities for interactive digital multimedia computing. ■

Acknowledgments

The anonymous reviewers provided helpful comments. NCR Corp. supported some of our work on interactive digital video. Gregory Fox and Mahesh Ursekar provided secretarial assistance. I extend special thanks to Deborah Hix, Edward Schwartz, and the many students working with us on digital video at VPI&SU.

References

1. *Managing Interactive Video/Multimedia Projects*, Educational Technology Publications, R. Bergman and T. Moore, eds., Prentice Hall, Englewood Cliffs, N.J., 1990.
2. *Learning with Interactive Multimedia*, S. Ambron and K. Hooper, eds., Microsoft Press, Redmond, Wash., 1990.
3. W. Mackay and G. Davenport, "Virtual Video Editing in Interactive Multimedia Applications," *Comm. ACM*, Vol. 32, No. 7, July 1989, pp. 802-810.
4. *CD-ROM, Vol. 2: Optical Publishing*, S. Ropiquet, J. Einberger, and B. Zoellick, eds., Microsoft Press, Redmond, Wash., 1987.

5. *The CD-ROM Handbook*, C. Sherman, ed., McGraw-Hill, New York, 1988.
6. J. Nadler and R. Wiesenbergs, "CD-ROM Hardware," in *The CD-ROM Handbook*, C. Sherman, ed., McGraw-Hill, New York, 1988, pp. 79-106.
7. J. Einberger and B. Zoellick, "High Sierra Group Format Description," in *CD-ROM, Vol. 2: Optical Publishing*, S. Ropiequet, J. Einberger, and B. Zoellick, eds., Microsoft Press, Redmond, Wash., 1987, pp. 195-216.
8. Philips International Inc., *Compact Disc-Interactive*, McGraw-Hill, New York, 1988.
9. D. Traub, "An Historical Perspective on CD-ROM," in *The CD-ROM Handbook*, C. Sherman, ed., McGraw-Hill, New York, 1988, pp. 17-50.
10. D. Davies, "Future Possibilities of CD-ROM," in *The CD-ROM Handbook*, C. Sherman, ed., McGraw-Hill, New York, 1988, pp. 209-239.
11. *Proc. First Int'l Workshop on Network and Operating System Support for Digital Audio and Video*, Tech. Report TR-90-062, D. Anderson et al., eds., Int'l Computer Science Inst., Berkeley, Calif., 1990.
12. R. Sansom and E. Cooper, "The Impact of Broadband Integrated Services on the Structure of Systems and Application Software," *Proc. First Int'l Workshop on Network and Operating System Support for Digital Audio and Video*, Tech. Report TR-90-062, Int'l Computer Science Inst., Berkeley, Calif., 1990.
13. D. Wybraniec, R. Cordes, and F. Stamen, "Support for Multimedia Communication in Future Private Networks," *Proc. First Int'l Workshop on Network and Operating System Support for Digital Audio and Video*, Tech. Report TR-90-062, Int'l Computer Science Inst., Berkeley, Calif., 1990.
14. M. Liebhold and E. Hoffert, "Toward an Open Environment for Digital Video," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 103-112.
15. R. Steinmetz et al., "Compound Multimedia Objects — Integration into Network and Operating Systems," *Proc. First Int'l Workshop on Network and Operating System Support for Digital Audio and Video*, Tech. Report TR-90-062, Int'l Computer Science Inst., Berkeley, Calif., 1990.
16. D. Anderson, R. Govindan, and G. Homay, "Design and Implementation of a Continuous I/O Server," *Proc. First Int'l Workshop on Network and Operating System Support for Digital Audio and Video*, Tech. Report TR-90-062, Int'l Computer Science Inst., Berkeley, Calif., 1990.
17. P. Momtahan and R. Kamel, "PX Connection Architecture," *Proc. First Int'l Workshop on Network and Operating System Support for Digital Audio and Video*, Tech. Report TR-90-062, Int'l Computer Science Inst., Berkeley, Calif., 1990.
18. H. Katseff et al., "An Overview of the Liaison Network Multimedia Workstation," *Proc. First Int'l Workshop on Network and Operating System Support for Digital Audio and Video*, Tech. Report TR-90-062, Int'l Computer Science Inst., Berkeley, Calif., 1990.
19. C. Yu et al., "Efficient Placement of Audio Data on Optical Discs for Real-Time Applications," *Comm. ACM*, Vol. 32, No. 7, July 1989, pp. 862-871.
20. A. Luther, *Digital Video in the PC Environment*, second edition, McGraw-Hill, New York, 1991.
21. J. Bottoms and L. Helgeson, "The First Step Toward Publishing on CD-ROM," in *The CD-ROM Handbook*, C. Sherman, ed., McGraw-Hill, New York, 1988, pp. 269-307.
22. A. Adkins, "Data Preparation and Pre-mastering," in *The CD-ROM Handbook*, C. Sherman, ed., McGraw-Hill, New York, 1988, pp. 343-396.
23. E. Hoffert and G. Gretsche, "The Digital News Systems at Educum: A Convergence of Interactive Computing Newspapers, Television, and High-Speed Networks," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 113-116.
24. B. Brewer, "Using Audio," in *CD-ROM, Vol. 2: Optical Publishing*, S. Ropiequet, J. Einberger, and B. Zoellick, eds., Microsoft Press, Redmond, Wash., 1987, pp. 169-183.
25. G. Doherty and R. Apperson, "Displaying Images," in *CD-ROM, Vol. 2: Optical Publishing*, S. Ropiequet, J. Einberger, and B. Zoellick, eds., Microsoft Press, Redmond, Wash., 1987, pp. 121-168.
26. A. Lippman, "Feature Sets for Interactive Images," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 92-101.
27. R. Bruno, "Compact Disc-Interactive," in *The CD-ROM Handbook*, C. Sherman, ed., McGraw-Hill, New York, 1988, pp. 131-185.
28. B. Haskell, "International Standards Activities in Image Data Compression," *Proc. Scientific Data Compression Workshop*, NASA Conf. Pub. 3025, NASA Office of Management, Scientific and Technical Information Division, Washington, DC, 1989, pp. 439-449.
29. M. Liou, "Overview of the px64 Kbit/s Video Coding Standard," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 59-63.
30. A. Fleig, "The EOS Data and Information System," *Proc. Scientific Data Compression Workshop*, NASA Conf. Pub. 3025, NASA Office of Management, Scientific and Technical Information Division, Washington, DC, 1989, pp. 73-83.
31. E. Fox, "Guest Editor's Introduction: Standards and the Emergence of Digital Multimedia Systems," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 26-29.
32. K. Rao and P. Yip, *Discrete Cosine Transform — Algorithms, Advantages, Applications*, Academic Press, London, 1990.
33. *Proc. Scientific Data Compression Workshop*, NASA Conf. Pub. 3025, H. Ramapriyan, ed., NASA Office of Management, Scientific and Technical Information Division, Washington, DC, 1989.
34. *Proc. DCC 91, Data Compression Conf.*, J.A. Storer and J.H. Reif, eds., IEEE CS Press, Los Alamitos, Calif., Order No. 2202, 1991.
35. M. Barnsley and A. Sloan, "Fractal Image Compression," *Proc. Scientific Data Compression Workshop*, NASA Conf. Pub. 3025, NASA Office of Management, Scientific and Technical Information Division, Washington, DC, 1989, pp. 351-365.
36. A. Blumer, "Noiseless Compression Using Non-Markov Models," *Proc. Scientific Data Compression Workshop*, NASA Conf. Pub. 3025, NASA Office of Management, Scientific and Technical Information Division, Washington, DC, 1989, pp. 367-375.
37. M. Cohn, "Performance of Lempel-Ziv Compressors with Deferred Innovation," *Proc. Scientific Data Compression Workshop*, NASA Conf. Pub. 3025, NASA Office of Management, Scientific and Technical Information Division, Washington, DC, 1989, pp. 377-391.
38. F. Sijlström and J. van der Meer, "CD-I Full-Motion Video Encoding on a Parallel Computer," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 81-91.
39. A. Lippman and W. Butera, "Coding Image Sequence for Interactive Retrieval," *Comm. ACM*, Vol. 32, No. 7, July 1989, pp. 852-860.
40. K. Harney et al., "The i750 Video Processor: A Total Multimedia Solution," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 64-78.
41. R. Gray, "Vector Quantization," *Proc. Scientific Data Compression Workshop*, NASA Conf. Pub. 3025, NASA Office of Management, Scientific and Technical Information Division, Washington, DC, 1989, pp. 205-231.
42. R. Baker, "AVLSI Chip Set Real-Time Vector Quantization of Image Sequences," *Proc. Scientific Data Compression*

Workshop, NASA Conf. Pub. 3025, NASA Office of Management, Scientific and Technical Information Division, Washington, DC, 1989, pp. 419-437.

43. G. Wallace, "The JPEG Still Picture Compression Standard," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 30-44.
44. D. Le Gall, "MPEG: A Video Compression Standard for Multimedia Applications," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 46-58.
45. H. Musman, "The ISO Audio Coding Standard," *Proc. Globecom 90*, IEEE, New York, 1990.
46. N. Borenstein, "Multimedia Electronic Mail: Will the Dream Become a Reality?" *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 117-119.
47. S. Newcomb, N. Kipp, and V. Newcomb, "HyTime: The Hypermedia/Time-Based Document Structuring Language," to appear in *Comm. ACM*, Vol. 34, No. 11, Nov. 1991.
48. G. Cockroft and L. Hourvitz, "Nextstep: Putting JPEG to Multiple Uses," *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, p. 45.
49. R. Phillips, "MediaView: A General Multimedia Digital Publication System," *Comm. ACM*, Vol. 34, No. 7, July 1991, pp. 74-83.
50. A. Luther, "Digital Video Interactive," in *The CD-ROM Handbook*, C. Sherman, ed., McGraw-Hill, New York, 1988, pp. 187-207.
51. Special section on interactive technologies, E. Fox, ed., *Comm. ACM*, Vol. 32, No. 7, July 1989.
52. J. Green, "The Evolution of DVI System Software," to appear in *Comm. ACM*, Vol. 35, No. 1, Jan. 1992.
53. M. Liebold, "Hypermedia and Visual Literacy," in *Learning with Interactive Multimedia*, S. Ambron and K. Hooper, eds., Microsoft Press, Redmond, Wash., 1990, pp. 99-110.



Edward A. Fox is associate director for research at the Computing Center and associate professor of computer science at Virginia Polytechnic Institute and State University. He directed the Virginia Disc series of CD-ROMs and Virginia Tech work on interactive digital video. His interests, research projects, and publications are in the area of information storage and retrieval, library automation, hypertext/hypermedia/multimedia, computational linguistics, CD-ROM and optical disc technology, electronic publishing, hashing, and expert systems.

Fox received a BS in electrical engineering from MIT and an MS and a PhD in computer science from Cornell University. He serves ACM as a member of the Publications Board, chairman of the Special Interest Group on Information Retrieval (SIGIR), and associate editor of *ACM Transactions on Information Systems*.

The author can be reached at the Department of Computer Science or the Computing Center, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061-0106; e-mail fox@vtopus.cs.vt.edu.

RESEARCH CHAIR IN SOFTWARE PERFORMANCE ENGINEERING

An NSERC-OCRI Industrial Research Chair in Software Performance Engineering has been established at Carleton University. Applications are invited from established researchers in university, industry or government. The Chair carries with it a substantial five-year grant for research into high-performance software for concurrent, parallel and distributed systems, including design, prediction and measurement problems, and oriented towards either throughput-oriented or time-delay oriented real time systems, or both. Applicants should be internationally known researchers with a research record in this area.

The Chairholder will be expected to cooperate with and to complement the work of the members of the Real-Time and Distributed Systems Group in the Department of Systems and Computer Engineering. The Group's laboratory is supported by NSERC and by industry. It is the centre of a project in Telecommunications Software Methods sponsored by TRIO as well as the Telecommunications Research Institute of Ontario. Its Director is Professor C.M. Woodside. Recent research in high-performance software includes analytic modelling, optimization, design notations and methods, languages and operating systems, simulation and performance instrumentation. Recent research has focused particularly on software for telecommunications, networks, and signal processing applications. A Chairholder with expertise in these areas, or in control or database/transaction-processing performance would be welcome.

The Real-Time and Distributed Systems Group has ten full-time members including both faculty and research staff and about twenty students divided equally between Masters and Doctoral programs. It has modern computing facilities and will shortly occupy enlarged laboratories in the new Minto Centre for Advanced Studies in Engineering. The Systems and Computer Engineering Department has a large graduate program at both the Masters and Ph.D. levels, and provides and excellent research environment.

The Chair is funded by the Cooperative Research program of the Natural Sciences and Engineering Research Council of Canada (NSERC) and by the Ottawa-Carleton Research Institute (OCRI) with participation through OCRI by Bell-Northern Research, Bell Canada, Digital Equipment Corporation, Gandalf Technologies, DY4 Inc., and by TRIO, the Telecommunications Research Institute of Ontario.

Applications should include a curriculum vitae and the names of three references. It should be directed to:

**Dr. S.A. Mahmoud, Chair
Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario, Canada, K1S 5S6
by 15 December 1991.**

In accordance with Canadian immigration requirements, this advertisement is directed in the first instance to Canadian citizens and permanent residents. Other qualified candidates are also invited to apply. Carleton University is committed to equality of employment for women, aboriginal peoples, visible minorities and disabled persons. Interested persons from these groups are encouraged to apply.

CARLETON • UNIVERSITY

SIEMENS



Siemens. The Brightest Force in R&D.

At Siemens, a \$40 billion global company, we fully recognize the value of Research and Development. That's why we invested over DM 7 billion in worldwide R&D efforts last year in support of our staff of over 400,000. In our centralized corporate R&D, located in Munich and Erlangen (Germany) and Princeton, NJ, we perform research and basic development in materials, products and systems technology in strategically important areas, as well as advise the board and the operating units in questions of research and development.

At our new 170-acre complex in Princeton, NJ, the U.S. center of corporate research for SIEMENS, we are focussing our strengths on three major research areas, namely, Learning & Knowledge Acquisition, Software Engineering and Imaging/Optical Computing.

Our active programs include joint projects with the leading U.S. universities and institutes in addition to the opportunity to share in the research and talent resources of Siemens global R&D organization. To qualify you should be ambitious and progressive with a Ph.D. in Computer Science or Electrical Engineering and demonstrated expertise in one of the following project areas:

Learning & Knowledge Acquisition

Basic and applied studies in the areas of Learning (Connectionist and AI), adaptive processes and knowledge acquisition.

Software Engineering

Research in methods and tools to improve present-day software engineering (design, testing and maintenance) and in breakthrough technologies to meet the challenges of tomorrow.

Image Processing & Analysis/ Optical Computing

Basic and applied research in a variety of areas including image management enhancement, segmentation, reconstruction, visual perception and, a new area, optical computing.

Let our commitments propel your achievements. For further information, please send your resume, indicating area of expertise and present base salary, to: Human Resources, Dept Comp, Siemens Corporate Research, Inc., 755 College Road East, Princeton, NJ 08540. Equal Opportunity Employer

Conceptual Modeling of Multimedia Documents

Carlo Meghini, Fausto Rabitti, and Costantino Thanos
Consiglio Nazionale delle Ricerche

Today, the demand is increasing for systems that can store, retrieve, and manipulate large volumes of multimedia documents such as structured collections of formatted data, text, graphics, images, and voice. This increasing demand has been caused by the rapid development of workstation technology in recent years that has made sophisticated workstations available at continuously declining costs. The proliferation of workstation-based office systems has expanded the production of multimedia documents.

However, the effectiveness of operational full-text retrieval systems is restricted by the built-in theoretical limits of existing retrieval models. In addition, multimedia documents introduce new requirements for retrieval. Most of the current text-information management systems ignore requirements for maintaining document structure and for storing nontext components such as graphics or images while still providing content-based access.

In this article, we present a new approach to the document-retrieval problem. The objective of this approach is to increase the efficiency and effectiveness of document-retrieval systems by exploiting the semantic contents of the documents. We also will illustrate an experimental system, the Multimedia Office Server (Multos),¹ which implements some of the document-model concepts described in the article.

Multos is an experimental system designed to enhance multimedia document retrieval. It is based on multilevel conceptual modeling principles for semantic structuring.

The document-retrieval problem

The problem of document retrieval is to identify the documents that contain information pertinent to user queries. An ideal function of a document-retrieval system might consist of selecting from the document base all documents that the requester will evaluate as sufficiently relevant. Conventional document-retrieval systems can be divided into two categories: keyword-based retrieval systems and full-text retrieval systems.

Keyword-based systems store textual documents and support document retrieval based on manually or automatically created document records containing predefined subject terms (descriptors). The document model these systems support is a simple one. A document is represented as a set of words (the keywords contained in the document record), each of which can be looked up in an index.

Full-text retrieval systems store textual documents and support document retrieval based on subject content. The document model these systems support is still a simple one. A document is defined as having content and attributes. The content is represented as a list of words, each of which can be looked up in an index. In some advanced systems, when lengthy documents are considered, the content is divided into chapters or sections. Each chapter or section is represented as a list of words. In addition, some attributes can be connected to chapters or sections. Thus, it is possible to have attributes at both the document and the chapter level.

Two parameters are widely used to measure the performance of document-retrieval systems: retrieval efficiency and retrieval effectiveness.

Important criteria for retrieval efficiency are system response time, user effort, and form of presentation. Keyword-based systems will perform better than full-text systems in terms of system response time. Indeed, according to one study,² the document-record size in keyword-based systems is 20 times shorter on average than the corresponding document. This means that, on average, the size of the document base to be scanned by the full-text retrieval software is 20 times larger than the size of the document base scanned by the keyword-based retrieval software.

There are two main drawbacks to keyword-based retrieval. First, choosing the terms to reflect the subject content of a particular document is not easy. Second, the use of only predefined descriptors to represent the document contents is a severe limitation.

The main drawback of full-text retrieval is that large portions of a document base must be scanned to search the specified documents. Indeed, full-text systems do not allow the user to specify "context areas" for confining the document search. As mentioned above, some of the most advanced com-

mercial systems introduce the concept of document "context area" by identifying it with a structural component of a predefined type, such as chapter or section. But this is still inadequate; to be successfully employed, we believe that context areas must be user-significant. Hence, the user should be able to structure documents through components that are meaningful to that user.

Retrieval effectiveness, the second parameter used to measure the performance of document-retrieval systems, is usually evaluated in terms of recall and precision. Recall is the portion of relevant documents actually retrieved from the document base, while precision is the portion of the retrieved documents found to be relevant to the user's needs.

To improve the recall and precision levels, recall-enhancing and precision-enhancing devices can be used to alter both query formulation and document representation. Recall-enhancing devices (for example, concept generalization) broaden the document and query identifiers. Precision-enhancing devices (for example, concept specialization and context area) make term identification more specific. How sophisticated the use of these techniques can be clearly depends on the sophistication of the document model supported by the system. These techniques, based on document semantic concepts, are orthogonal to other techniques (term truncation, term weighting, etc.), and thus could be employed in combination with them.

Multimedia documents place new requirements on content-based document-retrieval systems. Indeed, graphics, images, and voice messages carry information that enriches the textual content of the documents and could be usefully exploited by the user and the system in query formulation and the consequent document searching. More advanced existing document-retrieval systems allow multimedia information, such as graphics and images, to be stored and retrieved by the system; but this information is not used in searching. Some existing systems allow image information to be queried on the basis of manually inserted textual descriptors (captions) of the image content. These systems store multimedia document components as uninterpreted strings, making it difficult to capture their structure, properties, and interrelationships.

Conceptual modeling for documents

Next, we discuss conceptual document modeling basics and requirements.

Basic concepts. Conceptual modeling identifies high-level concepts and tools to solve the problems posed by the increasing complexity of new applications. In information systems, conceptual modeling formally began with the introduction of data models, notably the relational data model. About 10 years later, more powerful data-modeling tools, semantic data models,³ were proposed to allow a more adequate representation of the applications' semantics. Object-oriented semantic data models have now emerged as an effective tool for modeling modern database applications.

Typically, an object-oriented semantic data model views the reality as consisting of interrelated objects and offers abstraction mechanisms (particular relationships between objects) as structuring principles. The basic abstraction mechanisms are

- (1) *classification*, representing the "instance-of" relationship, by which objects with common properties are gathered into other objects called classes;
- (2) *aggregation*, representing the "part-of" relationship between an object and its components, by which an object can be seen as a compound of objects; and
- (3) *generalization*, representing the "is-a" relationship, by which classes can be organized into taxonomies, usually termed "is-a" hierarchies.

These mechanisms interact through inheritance, which can be categorized as inheritance of instances, by which an instance of a class is also an instance of its generalizations, and inheritance of properties, which allows the properties of a class to also be properties of its specializations. The second kind of inheritance can be immediately verified as a logical consequence of the first. Figure 1 shows the schema and data of a small self-explanatory semantic database.

Conceptual document modeling. The basic difference between a database and a document base is that a database is an

abstraction of the real world, that is, a collection of facts extracted from reality and encoded into the data structures provided by the database model. A document base is a collection of documents and, as such, not an abstraction; it contains the reality (that is, the documents) as it is. This view of a document base has confined conceptual document modeling to the development of abstract models that represent documents as complex objects composed of text, graphic, image, and voice components. Office document architecture (ODA)⁴ is an example. It is a model for document architecture that merges in a single framework the representation of the document structure (called logical model) and of the document presentation formats (layout model).

However, merely referring to the structure of a document is not sufficient to retrieve documents in a large document base. In general, a user's perception of a document can vary from document to document. For some documents, the perception can be based on appearance (layout); for others, it can be based on structure (logical); for still others, it can be based on content. Therefore, in addition to the logical and layout structures of the document, it is necessary to define a conceptual structure that will permit the representation of document contents.

We think of conceptual modeling as ranging between two extremes. On the one hand, there's the uninterpreted approach, in which documents are seen as sequences of words on which the system operates with no semantic knowledge. On the other hand, there's the semantic approach, in which documents are seen as sets of sentences of an artificial language, with an associated formal semantics.

Both the full-text and the keyword-based retrieval systems mentioned previously belong to the uninterpreted approach because they view documents as sequences of words, on which retrieval is carried out on a string-matching basis. The only difference is that keyword-based systems confine the search to restricted versions of the documents—that is, the records bearing the keywords. Statistical methods of information retrieval seem to lie in between the two approaches. However, from a semantic viewpoint, it is difficult to analyze the inference processes these systems perform.

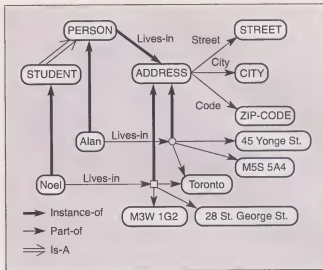


Figure 1. Sample semantic database.

Recently, there have been a number of attempts to approach the document-modeling problem from the other extreme.⁵ The basic idea is to apply natural-language-processing techniques to perform text indexing for further retrieval. So far, these attempts tend to be successful only on fragments of natural language. It could hardly be otherwise, given the intrinsic "complexity of natural language, the complexity of its processing, and the complexity of natural language texts in general."⁵ It is difficult to predict how such attempts would affect the performance of a system dealing with thousands of documents.

Two models have made the most progress in conceptual document modeling—Multos¹ and another proposed by Croft and Stemple.⁶ Both these models integrate the different views of a document—the layout view, the logical view, and the conceptual view—following different approaches.

Multilevel document presentation. An ideal document model should allow for a multilevel representation, including the conceptual, the logical, and the layout views of a document. More importantly, these views should be molded into a unique representational framework, thus giving each view the status of a first-class object of the model. This will permit the accommodation of standards and allow the user to manipulate all the representations of a document simultaneously and uniformly (for instance, when stating a query). Further-

more, the model should adopt the basic view of an object-oriented semantic data model, treating the various representations of documents and their components as objects and using abstraction mechanisms to organize the document base. Document evolution should also be a primary concern of the model, which must provide constructs to adequately handle the various stages of a document's existence.

Multimediality. Multimediality must be introduced into the logical view of documents by providing the document model with a primitive class for each different kind of media so the user can create multimedia objects as instances of these classes or of specializations of them. The objects so created would then be linked to each other and to the other objects via the aggregation mechanism, giving rise to multimedia complex objects. To allow queries on components other than formatted data and text, the contents of these components should be represented symbolically at the conceptual level. The symbolic representation of a nonformatted object is typically a set of interpretations, open to user inspection.

Querying. For the full exploitation of the expressive power of the model, the query language should satisfy these requirements:

- Treat objects and classes uniformly to allow the properties of classes to be

Figure 2.
Relationship between conceptual structure and ODA structures.

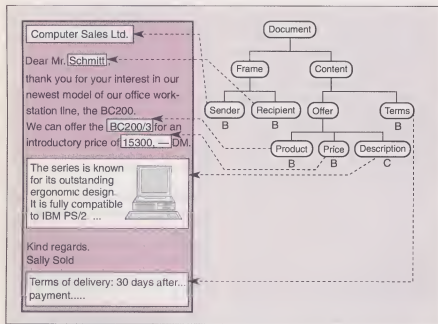
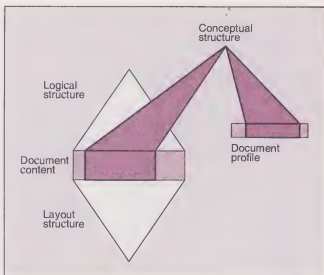


Figure 3. A document and its associated conceptual structure (B = basic, C = complex components).

queried the same way as the properties of objects. Examples of class properties are "number of documents in the class" and "maximum number of occurrences of the word 'semantics' within the documents in the class."

- Treat the definitional part (schema) and the factual part (data) of the document base uniformly, thus making it possible to query the definition of a class the same way as the properties of objects (for example, "the classes that

have a property of type text").

- Allow imprecise queries, with an incomplete specification of the structure and contents of the searched documents.

- Allow queries returning properties rather than documents; for instance, queries that ask the relation between two given objects.

Query processing. Processing the type of queries described in the previous sub-

section is much more complex than the usual query processing in traditional database systems or pure text retrieval systems. The main sources of complexity result from mixing data (objects) and metadata (classes) information in the queries, and from the specification of conditions on multimedia objects. The application of nonstandard techniques, including query completion and transformation by using schema information, is required to solve the former problem. Consideration of uncertainty factors is necessary to tackle the latter problem, since multimedia document retrieval is not an exact process.

Automatic classification of documents. The high level of structuring imposed by the model requires classifying a document at the conceptual, logical, and layout levels. The system should automatically perform as much of the classification process as possible. While this seems possible for the layout classification, problems arise with the logical and conceptual classifications. The former requires the system to match the logical structure of the incoming document with that defined by the classes modeling the logical view of the existing documents. Techniques developed in artificial intelligence for diagnosis problems seem to be appropriate for this task. The automatic conceptual classification is an open research problem, given the arbitrariness of a document's conceptual representation.

Multimedia document modeling and retrieval in Multos

The Multos project¹ can be considered one of the first attempts at designing and implementing a system for the storage and retrieval of multimedia documents based on a multilevel document model. The main objective of this section is to show how the ideas on conceptual document modeling are realized in the Multos system.

Multilevel document modeling in Multos. The Multos document model incorporates the standard ODA model specifying the logical and layout levels of document modeling. These levels support operations such as editing, formatting, presentation, and exchange.

The logical structure determines how logical components (that is, syntactic document components such as sections and paragraphs) are related to the document-content portions (that is, the basic data values in the documents such as text elements or image elements). The layout structure determines how layout components such as pages, frames, and blocks are related to the document-content portions.

The conceptual modeling level is defined outside ODA. The structure of a document in terms of its conceptual components is called the conceptual structure. Multos requires a document to contain a conceptual structure in addition to the ODA logical and layout structures. Figure 2 illustrates the relationship between the conceptual structure and the ODA structures in a document instance. The dark gray area in the figure indicates the portions of the document that can be addressed through the conceptual document structure and used in querying the document. The principle adopted is that, considering the conceptual structure as a tree, the conceptual components constituting the leaves of the tree are mapped to ODA logical components of the document at different levels of detail. At the lowest level, a conceptual component can be associated with a basic ODA object (that is, a simple document portion) or a portion of it. At a higher level, a conceptual component can be connected to a logical ODA object. In this case, a complex document portion (for example, composed of several text paragraphs and images) is denoted by a single document conceptual component.

Document types can be defined within this model, the rationale being that the structure specified within the type can be used to build efficient access methods and to aid in query formulation. However, since multimedia document structures tend to differ greatly from instance to instance, if a document type is defined as the complete structure of all the documents belonging to that type, conceptual document types would be obtained that match only one or a few document instances; thus, the advantages of type definition in document access and retrieval would be lost. For this reason, the concept of weak document type is introduced.

A weak type only partially specifies the structure of its instances, allowing components of unspecified type (called

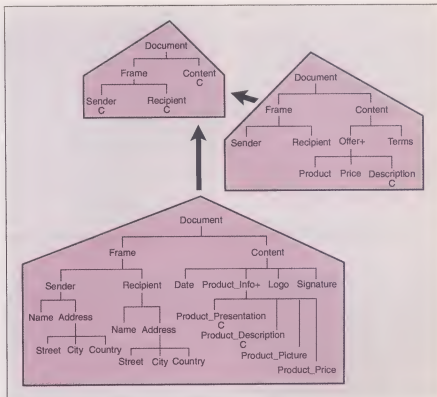


Figure 4. A hierarchy of conceptual document types (C = complex components, + = multivalued).

the complex component type) in a document type definition. A component of complex type can be specialized into some other type; the new document type so obtained can be considered a specialization of the previous document type. Thus, an is-a hierarchy can be constructed among weak types.

Figure 3 shows a simple example of a multimedia document containing text and images, along with the associated conceptual structure. Some of the basic conceptual components are connected to simple values, such as Sender, Receiver, Product, and Price; others to text portions, such as Terms; and still others to complex document components (that is, a logical ODA object), such as Description. Figure 4 shows an example of an is-a hierarchy of document types. Note that the leaf components of each document type can either be complex components (marked as C) or basic components. Observe that type Product_Presentation is obtained from Business_Document by specialization of the complex components Sender, Recipient, and Content, and type Product_Offer is obtained from type

Business_Document by a different specialization of the complex component Content. The "+" symbol attached to a component (for example, Offer in Product_Offer) means that it is multivalued (that is, at document level, several instances of this component can be present). The document in Figure 3 is of the Product_Offer type, with only one instance of the conceptual component Offer.

Multimediality. In Multos, the document components that have received the most attention have been graphic images (the restriction to graphic images rather than raster images has mainly been to simplify the analysis process, but the approach should be valid for both classes of images). The main conceptual problem in dealing with images derives from the difficulty of defining and interpreting their content exactly. It is difficult both to recognize the objects contained in an image and to determine and represent relationships between these objects, since they form structures that vary greatly from image to image.

In addressing the problem of image retrieval on large volumes of stored images, you need to accept some indeterminateness and then deal with the inaccuracy introduced by this feature. In Multos, the system designers decided to investigate an approach based on the mathematical theory of evidence, for the image recognition and description part, and on probabilistic theory, for the retrieval of previously analyzed images. For the representation of different degrees of similarity between objects, this combined approach allows the recognition, to a certain degree, of images, the objects contained in them, and the definition of classes of images/objects with unsharp boundaries.

Image retrieval is based on access structures representing the content of the images, obtained from a specific image analysis. In the image-analysis process, basic objects are recognized first, and then more complex objects are derived using domain-specific rules. In fact, images to be analyzed must belong to an application domain described in advance to the system by means of a specialized language.

Both pictorial and graphical images can be represented using a structural approach and interpreted according to the theory of evidence. The result of the image analysis is a set of interpretations, each constituted by the objects that have been recognized in the image. Each recognized object is characterized by the *belief* and the *plausibility* of recognition. The belief gives the total amount of belief committed to the object *O* after all evidence bearing on *O* has been pooled. The plausibility of *O* gives the extent to which the evidence allows one to fail to doubt *O* (that is, 1 - belief of the negation of *O*).

Figure 5 shows an example of symbolic representation of the image contained in the document of Figure 3. In the interpretation with higher belief, an image object IBM_like_PC has been recognized with belief 0.68 and plausibility 0.87. This object is composed of other objects, each recognized with a certain level of belief and plausibility.

Querying. The distinguishing features of the Multos query language are

- the capability to easily navigate through the document structure, realized by the concept of pathnames, used to identify document components. The

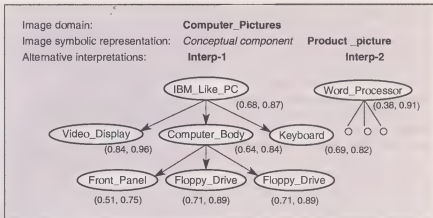


Figure 5. Result of image analysis (image symbolic representation).

FIND DOCUMENTS VERSION LAST WHERE

```
Document.Date > /1/1/1991/ AND
(*Sender.Name = "Olivetti" OR
*Product_Info CONTAINS "Olivetti") AND
(*Product_Description CONTAINS "Personal Computer%" OR
IMAGE *Product_Picture MATCHES
(DOMAIN Computer_Pictures OBJECTS IBM_like_PC (0.6 0.9)
WITH AT LEAST 2 Floppy_Drive (0.5 0.9) ) AND
(*Address.Country = "Italy" OR TEXT CONTAINS "Italy") AND
WITH *Company_logo;
```

Figure 6. An example of a multimedia query.

user can specify total paths (for example, Document.Frame.Sender.Address.City), identifying only one component, and partial paths, identifying several components (for example, Document* City denotes both Document.Frame.Sender.Address.City and Document.Frame.Recipient.Address.City).

- the specification of conditions on both the content and the conceptual structure of documents.

- the possibility of querying complex components. Such predicates apply to all the subcomponents that have a type compatible with the type required by the specific predicate (for example, certain predicates can be applied to numeric components; others, only to text components; and still others, to interpreted image components, etc.).

- the expression of predicates on the content of images.

Figure 6 shows an example of a multimedia query. With the query in Figure 6, the system looks for the last version

of all documents dated after January 1991 and having (1) the word "Olivetti" either as sender name or in the product information section (which is a nonleaf conceptual component); (2) the words "Personal Computer" (with a possible trailing 's') in the product description section or with the product picture that satisfies the image subquery (that is, an IBM-like PC object with at least two floppy-drive objects must have been recognized with at least the specified levels of belief and plausibility, after the analysis in the image domain Computer_Picture); (3) the word "Italy" either constituting the country in an address or contained in any part of the entire document; and (4) a company logo picture (no analysis is required on such a picture).

Query processing. The system designers considered the design of the Multos query processor⁷ important to show whether the ideas concerning the document model and the query language

```

FIND DOCUMENTS TYPE Product_Presentation VERSION LAST
WHERE
Document.Content.Date > /1/1/1991/ AND
(Document.Frame.Sender.Name = "Olivetti" OR
SOME Document.Content.Product_Info.Product_Presentation
CONTAINS "Olivetti"
OR SOME Document.Content.Product_Info.Product_Description
CONTAINS
"Olivetti")) AND
(SOME Document.Content.Product_Info.Product_Description
CONTAINS
"Personal Computer%" OR
SOME IMAGE Document.Content.Product_Info.Product_Picture
MATCHES
(DOMAIN Computer_Pictures OBJECTS IBM_like_PC (0.6 0.9)
WITH AT LEAST 2 Floppy_Drive (0.5 0.9))) AND
(Document.Frame.Receiver.Address.Country = "Italy" OR
Document.Frame.Sender.Address.Country = "Italy" OR
TEXT CONTAINS "Italy").

```

Figure 7. The results of a query transformation.

could be supported in an efficient implementation. Relevant aspects of query processing involve

- the transformation of imprecise queries;
- the use of different mechanisms to access components of different kinds (B-trees for formatted data, and specialized data structures for multimedia components);
- the complexity of type checking (due to the possibility for the user to specify partial paths or to omit any type specification in a query); and
- the implementation of the instance-of relation (an instance document is linked to the most specific type to which the conceptual structure of the document fits).

The previous example query in Figure 6 is transformed, during the type-checking phase, into the format in Figure 7. (This resulting query applies only to the Product_Presentation type; therefore, only the documents linked to this type and its eventual subtypes will be considered in processing the query.)

Automatic classification. The Multos system was focused on conceptual classification (that is, the construction of the document-conceptual structure). This can be done in three different ways:

manually, automatically, or semiautomatically. The user initiates the classification tools during or after the creation of documents within the document editor. Documents coming from outside are transformed into an internal format and classified automatically. Afterwards, the user has the possibility of refining the outcome of the classification process.

The automatic classification procedure in Multos is controlled by a document description at three different levels of abstraction based on the document model. The top level is the type hierarchy containing all defined document types, properties described by the second level — the conceptual structure definition. Finally, at the third level, predicates specify the content parts of documents relevant to the related conceptual component using syntactic, layout, linguistic, and background knowledge. This document description is compiled into classification rules that constitute the basis of the classification decisions.

We believe a multilevel semantic document model is an effective way to improve the efficiency and effectiveness of document-retrieval systems and that the Multos system is an important attempt

to apply semantic document-modeling techniques to obtain advances in document retrieval.

A further step toward more effective multimedia information retrieval would be the integration of conceptual modeling with a logic that will provide a powerful inference mechanism to support the retrieval process. ■

Acknowledgments

We gratefully acknowledge the contributions all participants made to the success of the Multos project. The Commission of the European Communities funded Multos from 1986 to 1990 as part of ESPRIT (European Strategic Program for Research in Information Technology).

References

1. *Multimedia Office Filing: The Multos Approach*, C. Thanos, ed., North-Holland, Amsterdam, 1990.
2. D.C. Blair and M.E. Marron, "An Evaluation of Retrieval Effectiveness for a Full-Text Document Retrieval System," *Comm. ACM*, Vol. 28, No. 3, Mar. 1985, pp. 289-299.
3. J. Peckham and F. Maryanski, "Semantic Data Models," *ACM Computing Surveys*, Vol. 20, No. 3, Sept. 1988, pp. 153-190.
4. W. Horak, "Office Document Architecture and Office Document Interchange Formats: Current Status of International Standardization," *Computer*, Vol. 18, No. 10, Oct. 1985, pp. 50-60.
5. Special issue on natural language processing and information retrieval, *Information Processing and Management*, A.F. Smeaton, ed., Pergamon Press, Vol. 26, No. 1, Jan. 1990.
6. W.B. Croft and D.W. Stemple, "Supporting Office Document Architectures with Constrained Types," *Proc. ACM SIGMOD*, 1987, pp. 504-509.
7. E. Bertino, S. Gibbs, and F. Rabitti, "Query Processing in a Multimedia Document System," *ACM Trans. Office Information Systems*, Vol. 6, No. 1, Jan. 1988, pp. 1-41.



Carlo Meghini has been a researcher at Istituto di Elaborazione dell'Informazione (IEI), Consiglio Nazionale delle Ricerche, Pisa, since 1984. His research interests include conceptual modeling and knowledge representation. He formerly served as a visiting scientist at the University of Toronto, working on projects within the Knowledge Representation Group, and led the IEI team in the ESPRIT Tools for Designing Office Information Systems project.

Meghini graduated with a degree in computer science from the University of Pisa in 1979.



Fausto Rabitti has been a staff researcher at Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, since 1982. His main research interests are in the areas of database management and information retrieval systems, with particular emphasis on new research fields such as multimedia information retrieval and object-oriented databases. He previously served as a research fellow at the Computer Systems Research Institute of the University of Toronto, working in the area of distributed office information systems; was a visiting scientist working in the Orion object-oriented database system project at MCC in Austin, Texas; and worked on the ESPRIT Multis project.

Rabitti received the Laurea degree in computer science at the University of Pisa in 1976.



Costantino Thanos has worked at Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, since 1970 and is currently head of the institute's Advanced Information Processing Techniques Department. He is in charge of CNR teams participating in a number of ESPRIT projects. His research interests include database systems, multimedia information retrieval, distributed databases, and office systems.

Thanos graduated with a degree in electronic engineering from the University of Pisa in 1970.

Readers can contact the authors at Istituto di Elaborazione dell'Informazione (Institute for Information Processing), Consiglio Nazionale delle Ricerche (Italian National Research Council), Via S. Maria, 46 - Pisa, Italy.

SHPCC '92 Call for Papers

Scalable High Performance Computing Conference

April 26-29, 1992, Williamsburg Hilton, Williamsburg, VA.

Robert Voigt
General Chair
ICASE

Program Committee
Joel Saltz, Chair
ICASE

Geoffrey Fox
Syracuse Univ.
Dennis Gannon
Indiana Univ.
Andrew Grimshaw
Univ. of Virginia
Michael Heath
Univ. of Illinois
Lennart Johnson
Thinking Machines
Robert Martino
NIH
Paul Messina
Cal. Tech.
Dan Reed
Univ. of Illinois
Manuel Salas
NASA Langley
Horst Simon
CSC

The Scalable High Performance Computing Conference (SHPCC) is the successor to the Distributed Memory Computing Conference series. We define a scalable high performance architecture as an architecture that is likely to be capable of delivering a teraflop sustained performance in the relatively near future. SHPCC will focus on software being developed to make it possible to effectively exploit the coming generations of these architectures and on applications that require teraflop computational rates.

We invite papers from researchers who are developing or evaluating software tools such as compilers, programming environments, and debuggers along with tools to monitor and tune performance. Papers are also solicited from those who have developed and analyzed multiprocessor applications codes. We plan to place particular emphasis on two general application areas. The first is that of computational fluid dynamics and its uses in problems that arise in the design of aircraft, spacecraft and automobiles. The second application area consists of molecular dynamics and its uses in the pharmaceutical and chemical industries.

The conference will begin with one day of tutorials. The two and one half day technical program that follows will consist of four invited talks, two parallel tracts of half hour talks, and a small poster session. We invite researchers to submit a four page extended abstract in hard copy, LaTeX, or Troff to the address below by November 15, 1991. The abstracts will be carefully reviewed by the program committee and the authors will be notified by early January. The proceedings will be published by IEEE and will be available at the time of the conference. Final papers will be due approximately March 1, 1992.

Ms. Emily Todd, ICASE, Mail Stop 132-C, NASA Langley Res. Ctr., Hampton, VA 23665
email: emily@icase.edu



COMPASS



BCS
THE SOCIETY OF
INFORMATION
SYSTEMS
ENGINEERING

CALL FOR PAPERS

Seventh Annual Conference on Computer Assurance

The Seventh Annual Conference on Computer Assurance (COMPASS), sponsored by the Institute of Electrical and Electronic Engineers and IEEE Aerospace and Electronic Systems Society, in cooperation with the British Computer Society, will be held at the National Institute for Standards and Technology in Gaithersburg, Maryland, USA, June 15-18, 1992. The purpose of this conference is to bridge the gap between emerging technology for computer assurance from research laboratories into industrial computer systems development. Papers may present original research on theoretical aspects and applications of technology to assured computing, or may be reports detailing experiments, evaluations, and open problems in the use of new technologies for computer assurance. Typical but not exclusive topics of interest include:

- Models and modelling (process, mathematical, and requirements models)
- Formal approaches (proofs of correspondence, formal specifications, and IV&V)
- Experiences with assurance (illustrative examples from communications, energy, financial, medical, military, transportation, and other types of systems)

PAPER SUBMISSION:

Authors are requested to send five single-sided copies of their papers (not exceeding 7,500 words) to the program chair by **January 10, 1992**. If available, an electronic mail address for the contact author should be included. Papers submitted simultaneously to another conference with published proceedings are disqualified. Papers will be refereed by the Program Committee and will be returned with comments. Accepted papers will be published in the proceedings.

IMPORTANT DATES:

Papers due:	January 10, 1992
Notification of acceptance:	March 7, 1992
Camera-ready paper due:	April 1, 1992
Conference:	June 15-18, 1992

Additional information about the COMPASS '92 can be obtained from the General Chair. All inquiries concerning paper submissions should be addressed to the Program Chair.

GENERAL CHAIR:

Robert Ayers
ARINC Research Corporation
2551 Riva Road
Annapolis, MD 21401 USA
voice: (301)266-4741
fax: (301)266-4040

PROGRAM CHAIR:

Dr. Edgar H. Sibley
Department of Information and Software Systems Engineering
George Mason University
4400 University Drive
Fairfax, VA 22030-4444
(703)993-1640, or esibley@gmuvox.gmu.edu



CALL FOR PAPERS
Fifth International Conference
on Architectural Support for
Programming Languages
and Operating Systems

Rapid advances in hardware technology, compilers, operating systems, and computer architecture provide a rich environment in design alternatives. The fifth ASPLOS conference focuses on the interaction of these technologies.

Full papers (approximately 5000 words) are solicited for, but not limited to, these topics:

- Interaction of operating systems with architecture (e.g., architecture support for O.S. features or architecture-influenced O.S. designs)
- Interaction of compilers and languages with architectures (e.g., compiling for parallel processors or architecture support for languages)
- Comparative (especially quantitative) analysis of architectural design choices (e.g., superscaler vs. VLIW vs. vector)
- Architectures for high-speed communications or interconnection networks
- Support for I/O and studies of I/O subsystems
- Performance evaluation of experimental and novel systems
- Impact of future technologies on programming languages, operating systems, or compilers.

We also solicit proposals for presentations of innovative systems that will be announced before the conference but not by the deadline for paper submissions. Contact the program chairman for details.

General Chair
Barry Flahive
Hewlett-Packard
300 Apollo Drive
Chelmsford, MA 01824
(flahive@apollo.hp.com)



SIGARCH
SIGPLAN
SIGOPS

October 12-15, 1992
Boston, Massachusetts

Sponsored by the ACM
in cooperation with the
IEEE Computer Society



TCMM
TC VLSI
TC OS

Accepted papers will be published in the Proceedings, which will be distributed at the conference and published as an issue of the ACM SIGARCH, SIGOPS, and SIGPLAN newsletters. Papers of particular merit will also be forwarded for publication in *ACM Transactions on Computer Systems*.

Because the identity of the authors will not be revealed to the referees, **authors' names and affiliations should appear only on a cover sheet together with the paper's title. Authors should also avoid citations that make their identities obvious.**

Five copies of submitted papers should be sent by **March 17, 1992**. Authors will be notified by **June 5, 1992**. Accepted papers must be printed on special forms and received by the program chair by **July 13, 1992**.

Submissions will be read by members and designated reviewers. Program committee members are:

Hank Levy, *University of Washington (chair)*
Anant Agarwal, *MIT*
Anita Borg, *DEC Western Research Lab*
David Cheriton, *Stanford University*
Doug Clark, *Digital Equipment Corporation*
David Callahan, *Tera Computer Corporation*
David Ditzel, *Sun Microsystems*
Susan Eggers, *University of Washington*
Thomas Gross, *Carnegie Mellon University*
Randy Katz, *University of California, Berkeley*
David May, *INMOS Limited*
Mary Lou Soffa, *University of Pittsburgh*
Guy Steele, *Thinking Machines Corporation*
John Wilkes, *Hewlett-Packard Laboratories*

Tutorials will be held as part of the conference. If you are interested in teaching a tutorial or feel the need for one in a particular field, please submit a proposal with a brief syllabus.

Program Chair
Hank Levy
Dept. of Computer Science and Engineering
University of Washington
Seattle, WA 98195
(levy@cs.washington.edu)

Automating the Generation of Coordinated Multimedia Explanations

Steven K. Feiner and Kathleen R. McKeown
Columbia University

Sometimes a picture is worth the proverbial thousand words; sometimes a few well-chosen words are far more effective than a picture. Pictures often describe objects or diagram physical actions more clearly than words do. In contrast, language often conveys information about abstract objects, properties, and relations more effectively than pictures can. Pictures and language used together can complement and reinforce each other to enable more effective communication than can either medium alone. In this sense, multimedia information systems may greatly increase effective communications.

Fortunately, technical advances are beginning to reduce the cost of hardware for computer-based multimedia and hypermedia. First-generation authoring facilities let users create presentations that include text, graphics, animation, and video. Regardless of the basic functionality or interface provided, however, multimedia authoring systems require authors to possess even more skills than do single-medium authoring systems. Not only must authors be skilled in the conventions of each medium, but they must also be able to coordinate multiple media in a coherent presentation, determining where and when to use different media, and referencing material in one medium from another. Furthermore, since the presentation must be authored in advance, the ways in which it can be customized for an individual user or situation are limited to those built in by the author.

To overcome the disadvantages of this predesigned authoring, we have developed an experimental test bed for the automated generation of multimedia explanations. COMET (Coordinated Multimedia Explanation Testbed)¹ has as its goal the coordinated, interactive generation of explanations that combine text and three-dimensional graphics, all of which is generated on the fly.

In response to a user request for an explanation, COMET dynamically determines the explanation's content using constraints based on the type of request, the information available in a set of underlying knowledge bases, and information about the user's background and goals. Having determined *what* to say, COMET also determines *how* to express it at the time of generation. The pictures and text that it uses are not "canned": COMET does not select from a database of conventionally authored text, preprogrammed graphics, or prerecorded video.

We developed an explanation system to overcome the disadvantages of conventional authoring in multimedia applications. COMET not only determines what to say, but how to say it.

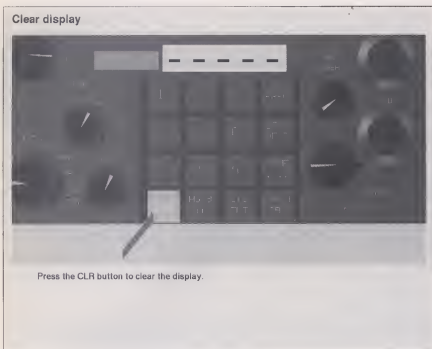


Figure 1. COMET's directions for clearing the radio display.

Instead, COMET decides which information should be expressed in each medium, which words and syntactic structures best express the portion to be conveyed textually, and which graphical objects, graphical style, and picture structure best express the portion to be conveyed graphically. COMET's text and graphics are created by separate *media generators*, each of which can communicate with the other.

We first provide a brief overview of COMET's domain and architecture. Then we focus on the specific ways in which COMET can coordinate its text and graphics. *Coordination* begins with the choice of media in which specific information is communicated. For example, an object's complex shape may be shown in a picture, rather than described in text, while a sentence may describe a causal relation between several actions involving the object. Coordination also means applying knowledge about what information is expressed in text to influence the generation of graphics, and vice versa. Thus, the graphics generator may use the fact that a causal relation is being communicated in text to determine how it depicts other information, even though the relation itself is not depicted. Finally, coordination means using knowledge about how information is expressed in other media in deci-

sion-making. For example, if the graphics generator shows the location of an object by highlighting it, the text generator can refer to "the highlighted object."

Overview

Much of our work on COMET is being done in a field maintenance and repair domain for a military radio receiver-transmitter. When provided with a set of symptoms, COMET generates multimedia explanations that instruct the user in how to carry out diagnostic tests. The user interacts with COMET by means of a simple menu and can initially choose to request instructions for a specific procedure or to invoke the diagnostic component. In the latter case, an underlying expert system is called to determine the problems that the radio is experiencing and to identify their causes.

The user selects symptoms from a menu. If the expert system decides that a set of diagnostic tests must be run to determine the cause of the problem, it calls the explanation component to tell the user how to carry out these tests. Explanations consist of one or more steps that are presented in a series of displays. Although our emphasis thus far has been on generating explanations, rather than on navigating through them, COMET's menu interface also provides rudimentary facilities for exploring the explanation by paging for-



Figure 2. Two steps from COMET's explanation for installing the holding battery.

ward and backward through its steps. It can also access steps by name.

Figure 1 shows COMET's explanation for clearing the radio's display. Figure 2 shows the beginning of COMET's multistep explanation for installing a new "holding battery." (The holding battery provides power for the radio's memory when the main battery has been removed.) In these first two steps, the user is instructed to turn the radio upside down and remove the cover plate from the battery compartment. Replacing the holding battery is the first of a series of actions that COMET instructs the user to perform in the course of troubleshooting loss of radio memory, a symptom that the user selected from COMET's menu.

System organization. COMET consists of a set of parallel processes that cooperate in the design of an explanation, as shown in Figure 3. On receiving a request for an explanation, the content planner uses text plans, or schemas,² to determine which information from the underlying knowledge sources should be included in the explanation. COMET uses three different domain knowledge sources: a static representation of domain objects and actions encoded in the Loom knowledge repre-

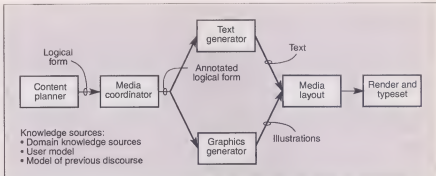


Figure 3. System architecture.

sentation language,³ a diagnostic rule base, and a detailed geometric knowledge base needed for graphics generation. It also maintains a user model and a model of the previous discourse. These knowledge sources are used by all system components to construct the explanation, not just by the content planner. Consequently, they are shown separately at the bottom of the figure without arrows to each module.

The content planner produces the full content for the explanation, represented as a hierarchy of logical forms⁴ (LFs), as explained in the sidebar titled "Unification in COMET." The LFs then are

passed to the media coordinator. This component refines each LF by annotating it with directives that indicate which portions are to be produced by each of the media-specific generation systems. The text generator and graphics generator each process the same LFs, producing fragments of text and graphics that are keyed to the LFs they instantiate. The output from both generators is processed by the media-layout component, which formats the final presentation for the low-level rendering-and-typesetting software.

COMET's major components run in parallel on up to five networked workstations. Text and menus are displayed through the X Window System, while 3D shaded graphics are rendered by Hewlett-Packard's Starbase graphics package. Each example shown in the figures takes approximately 10-20 seconds to generate and display following the initial user request.

One main feature of COMET's architecture is the use of the LF as a type of blackboard facility. (A blackboard⁵ is a central repository in which a system component can record its intermediate decisions and examine those of other components.) Each component reads and annotates its LF, continually enriching it with further decisions and finer specifications until the explanation is complete. Annotations include directives (like the media coordinator's choice of medium) and details about how a piece of information will be realized in text or graphics (like the proper verb to convey an action). While the annotated LF serves as a blueprint for the final explanation, it also allows for communication between media-specific components. For example, when deciding which expressive possibilities best con-

(b)

Install the new holding battery. Step 2 of 7



Step 2:
Remove the holding battery cover plate:
Loosen the captive screws and pull the holding battery cover plate off of the radio.

Unification in COMET

COMET uses FUF, an efficient extended version of Functional Unification Grammar,⁶ for media coordination. FUF also performs two text-generation tasks (selecting words and generating syntactic structure) and part of graphics generation (mapping the LFs to a communicative goal language supported by the graphics generator). Each component has its own "grammar" that is unified nondeterministically with the LF to annotate it with directives or further specifications. The result is a cascaded series of FUF grammars, each handling a separate task.

In FUF, both the input LF and the task grammar are represented using the same formalism, a set of attribute-value pairs. A value can be an atomic symbol or, recursively, a set of attribute-value pairs. For example, consider the following fragment of an input LF:

```
(substeps
  ((process-type action)
   (process-concept c-push)
   (roles (...)) ...))
```

This fragment contains a single attribute, *substeps*, whose value is a set of attribute-value pairs. It contains three subattributes. The first, *process-type*, specifies the type of substep process as an action. The second, *process-concept*, indicates that the specific action is a knowledge-base concept called *c-push* that represents the action of pushing. The third, *roles*, specifies the actors and objects that participate in the action. The value of the *roles* attribute is a set (not shown here). Additional attribute-value pairs occur in a full LF.

Annotation is accomplished by *unifying* the task grammar with the input and is controlled by the grammar. For each attribute in the grammar that has an atomic value, any corresponding input attribute must have the same value. (Technically, it must have a compatible⁶ value. As one example, when the grammar attribute has the value *Any*, the input attribute can have any value.) When the values are different, unification fails. When the attributes match and both values are sets, unification is applied recursively to the values, and the result replaces the input value. When the input LF does not contain the grammar attribute, the attribute and its value are added to the input. Any attributes that occur in the input but not in the grammar remain in the input after unification. Thus, unification matches only the relevant subsections of the input. Note that unification is similar to set union, since enriched attribute-value pairs from both input and grammar are merged.

A fragment of the media coordinator's grammar states that an action should be realized in both graphics and text:

```
((process-type action) ;; If process is an action
 (media-graphics yes) ;; Use graphics
 (media-text yes) ;; Use text
 ...))
```

On unifying this fragment of the grammar with the value of the *substeps* fragment of the input LF, FUF first checks whether the attribute *process-type* occurs in the input. Since it does and the value following it is *action*, FUF now checks whether the attribute *media-graphics* occurs in the input. Since it does not, FUF adds to the input the attribute-value pair (*media-graphics yes*), a directive indicating that the action is to be realized in graphics. Similarly, it adds the directive (*media-text yes*), specifying that the action is also to be realized in text. Thus, the first attribute-value pair is used as a test for this grammar portion. When it matches the input (the input LF describes an action), the input is annotated with the remaining attribute-value pairs. If the input LF had contained a different *process-type* (like *abstract*), unification with this portion of the grammar would have failed and FUF would have attempted unification with a new portion.

In most uses of the media coordinator grammar, a fragment is recursively applied to small nested segments of the input LF at least once. For example, the input LF can contain a causal relation, with two roles, both of which are actions. In this case, the grammar fragment as shown matches each role. A different fragment matches the causal relation (one that annotates it to be realized in text only). The roles of the actions are annotated recursively, as are their modifiers.

vey the specified content, a media generator can examine decisions made in other media by reading their annotated LFs and use that information to influence its own choices. COMET uses a single mechanism, FUF (Functional Unification Formalism), to make annotations throughout the system. This allows for additional bidirectional interaction between COMET's components through the use of unification, as described in the sidebar. Before describing coordination in more detail, we briefly discuss some of COMET's key components, focusing on their individual capabilities.

Media coordinator. This component performs a fine-grained analysis of an input LF to decide whether each portion should be realized in either or both media. After conducting a series of informal experiments and a survey of literature on media effectiveness, we distinguished six different types of information that can appear in an LF. We have categorized each type as to whether it is more appropriately presented in text or graphics. We use graphics alone for location and physical attributes, and text alone for communicating abstract actions and expressing connectives that indicate relationships among actions, such as causality. Both text and graphics represent simple and compound actions. The media coordinator has a FUF grammar that maps these information types to media.

Figure 4 shows a representative portion of the annotated LF produced by the media coordinator for Figure 1. The part of the LF in roman font was generated by the content planner, while the annotations added by the media coordinator are in boldface. This LF specifies a single substep and its effect, where the substep is a simple action (*c-push*) and the effect is also a simple action (*c-clear*). The *c-push* substep has one role (the medium, *c-button-clr*), and it also specifies that the location and size of the button should be included.

Figures 1 and 4 illustrate the fine-grained division of information among media. For example, location information is portrayed in graphics only, while the actions are realized in both text and graphics. In contrast, other information in the LF is communicated only in text, such as the causal relation between pushing the button and clearing the display.

Text generator. COMET's text generator² realizes the LF segments it has been assigned in text. It must determine both the number of sentences needed to realize the segments and their type (compound, simple, declarative, or imperative). It must select verbs to express LF actions, and nouns and modifiers to refer to LF objects. Finally, it must construct the syntactic structure for each sentence and linearize the resulting tree as a sentence.

The text generator divides into two modules that carry out these functions: the *Lexical Chooser* and the *Sentence Generator*. The Lexical Chooser selects the overall sentence type and the words, while the Sentence Generator produces individual sentences. Both modules are implemented using FUF.

The text generator can select words based on a variety of underlying constraints. This enables it to use a number of different words for the same LF concepts, depending on the context. The result is a wider variety of more appropriate output. COMET can use constraints from the underlying knowledge base, from previous discourse, from a user model, and from syntax. For example, for the knowledge-base concept c-install, the text generator can use "install," "reinstall," or "return." It makes a choice based on previous discourse (that is, what it has already told the user). Thus, after the user has installed the new holding battery, COMET instructs the user to remove the radio's main battery to check the new holding battery's functionality. At this point, COMET uses the previous discourse to select the verbs "reinstall" and "return" when instructing the user to put the main battery back in the radio. If the user had not been previously instructed to remove the battery, COMET would have selected the verb "install."

COMET can also avoid words that the user does not know. For example, it

```
((cat lf)
 (directive-act substeps)
 (substeps
  [((process-type action)
    (process-concept c-push)
    (mood non-finite)
    (speech-act directive)
    (function ((type substeps)
               (media-text yes)
               (media-graphics no)))
    (roles
     (medium
      ((object-concept c-button-clr)
       (roles
        ((location ((object-concept c-location)
                    (media-graphics yes)
                    (media-text no)))
         (size ((object-concept c-size)
                (media-graphics yes)
                (media-text no))))))
        ...
       ))
      (cat lf)
      (media-graphics yes)
      (media-text yes))))
 (effects
  [((process-type action)
    (process-concept c-clear)
    (mood non-finite)
    (function ((type effects)
               (media-text yes)
               (media-graphics no)))
    (speech-act assertive)
    (roles
     ((agent
      ((object-concept c-display)
       (roles
        ((location ((object-concept c-location)
                    (media-graphics yes)
                    (media-text no)))
         (size ((object-concept c-size)
                (media-graphics yes)
                (media-text no))))))
        ...
       ))
      (cat lf)
      (media-text yes)
      (media-graphics yes))))])
```

Figure 4. Logical form for Figure 1.

generates "Make sure the plus lines up with the plus" instead of "Check the polarity" when describing battery installation to a user not familiar with the word "polarity." The novel use of FUF to represent the lexicon efficiently provides a variety of different interacting constraints on word choice.

Graphics generator. IBIS (Intent-Based Illustration System)³ generates illustrations designed to satisfy the communicative goals specified in the annotated LFs that it receives as input. The communicative goals that IBIS currently supports include showing absolute and relative locations of objects, physical properties (such as size, shape, material, and color), state (such as a knob setting), change of state (such as the change in a knob setting), and a variety of actions (such as pushing, pulling, turning, and lifting). In designing an illustration, IBIS controls all aspects of the picture-making process: the objects included and their visual attributes, the lighting specification, the graphical style used in rendering the objects, the viewing specification, and the structure of the picture itself.

IBIS uses a generate-and-test approach. The IBIS rule base contains at least one design rule for each communicative goal that can appear in an input LF. Each design rule invokes a set of stylistic strategies that specify high-level visual effects, such as highlighting an object. These strategies are in turn accomplished by still lower level rules that realize the strategies. The lower level rules create and manipulate the graphical depictions of objects included in the illustration and modify the illustration's lighting specification, viewing specification, and rendering information.

For example, IBIS uses a combination of techniques

to portray the location of the button in Figure 1, as requested in the LF of Figure 4. It selects a viewing specification that (1) locates the button panel centrally in the illustration, (2) makes additional, surrounding context visible, and (3) ensures that both the object and context are recognizable. It highlights

the button by modifying the intensity of the lights that illuminate the objects in the illustration.

IBIS rules evaluate the success of each task that it performs. This is important because of the complex interactions that can occur in an illustration. Consider object visibility. Each object may be obscured by or obscure other objects. IBIS must determine whether visibility constraints are violated and address these by modifying the illustration. If an IBIS strategy doesn't succeed, it can backtrack and try another one. For example, if illuminating an object doesn't make it brighter than surrounding objects, IBIS can try to decrease the intensity of the lights illuminating the surrounding objects.

Media coordination

A multimedia explanation system must coordinate the use of different media in a single explanation. It must determine how to divide explanation content between different media such as pictures and text. Moreover, once the content has been divided, the system must determine how material can be generated in each medium to complement that of the other media.

A few researchers are addressing the automated generation of coordinated multimedia explanations with emphasis on how the media can complement each other. Integrated Interfaces⁸ produces US Navy briefing charts, using rules to map objects in the application domain (a database of information about ships) into objects in the presentation. Sage⁹ explains how and why quantitative models change over time, while Wip¹⁰ explains physical actions like those of COMET's domain.

Integrated Interfaces and Sage operate in a two-dimensional world of charts and graphs, and do not address the problems of describing objects and actions in 3D. Integrated Interfaces also uses many design rules specific to the particular kind of briefing chart it produces. Although Wip also emphasizes tight media coordination in application to 3D domains, its content planner takes an incremental approach. Each piece of information to be communicated is assigned sequentially to its generators. Evaluations of potential success are returned to the planner to help determine media assignments even before enough

information is provided for an entire sentence or illustration.

In contrast, COMET provides its media generators with more information at a time in a common LF that describes what the other generators have been assigned. The LF can also be enriched with accomplishments of the other generators. Thus COMET gives its media generators more context from which to work in making their initial decisions, while still allowing feedback.

Here we focus on two aspects of media coordination in COMET. First, we show how the use of a common content-description language allows for more flexible interaction between media, making it possible for each generator to query and reference other generators. By passing the same annotated description of what is to be described to each generator, we permit each generator to use information about what the other generators present to influence its own presentation. Then, we show how bidirectional interaction between the media-specific generators is necessary for certain kinds of coordination. Bidirectional interaction allows COMET to generate explanations that are structurally coordinated and that contain cross-references between media.

Common content description

All components in COMET following the content planner share a common description of what is to be communicated. Just as modules accept input in the same formalism, they can also annotate the description as they carry out its directives. This design has the following ramifications:

- It lets text and graphics influence each other.
- Communicative goals are separated from the resources used to carry them out.
- It provides a mechanism for text and graphics generators to communicate.

Mutual influence. Since both the text generator and graphics generator receive the same annotated content description as input, each knows which goals are to be expressed in text, in graphics, or both. Even when a media-specific generator does not communi-

cate a piece of information, it knows that the information is to be conveyed to the user; thus, it can use this knowledge to influence its presentation. Consider a portion of the explanation that COMET generates to instruct the user in how to install the holding battery. The second step of the explanation (Figure 2b) was generated from a complex LF that consists of one goal (to remove the holding battery cover plate) and two complex substeps that carry out that goal. As Figure 2b illustrates, the media coordinator determines that the goal is to be generated only in text ("Remove the holding battery cover plate:") and that the substeps are to be shown in both media.

Although IBIS depicts only the substeps of the LF, it receives the entire annotated LF as input. Since it receives the full LF, and not just the pieces assigned to graphics, IBIS knows that the actions to be depicted are steps that achieve a higher level goal. Although this goal is not itself realized in graphics, IBIS uses this information to create a composite illustration.⁷ This type of illustration consists of an integrated set of pictures that work together to achieve a common set of goals that cannot be accomplished in a single "simple" illustration. In this case, IBIS rules do not include any satisfactory way to show the radio with its cover plate and captive screws in different positions in one illustration.

If IBIS were to receive only the substeps, it would have no way of knowing that the substeps are being described in relation to a higher level goal. It may end up producing two separate illustrations, just as it does for each simple LF, such as that shown in Figure 2a. Thus, information conveyed in the explanation as a whole, but not in graphics, influences how IBIS depicts other information.

Separation of goals from resources. Because we are using a common content-description language, content must be specified at a level that is appropriate for all generators. We have found that by expressing content as a combination of communicative goals and the information needed to achieve these goals, each generator can select the resources it has at hand for accomplishing its assigned goals. In text generation, this means the selection of specific syntactic or lexical resources (using passive

voice to indicate focus, for example). In graphics generation, it means the selection of a conjunction of visual resources (modifying an object's material and the lights that illuminate it to highlight it, for example).

Consider again the explanation shown in Figure 1 and its associated annotated LF in Figure 4. The main goal of the first part of the LF is to describe an action (c-push) and its role (medium). Subgoals include referencing an object (for example, c-button-clr, the clear button) and conveying its location and size. IBIS and the text generator use different resources to achieve these goals. For example, the text generator selects a lexical item, the verb "press," to describe the action. "Press" can be used instead of other verbs because of the characteristics of the medium, c-button-clr. If the medium were a slider, a verb such as "push" or "move" would be required. In contrast, IBIS uses a *metaobject*, an object that does not itself represent any of the objects in the world being depicted. In this case, the metaobject is an arrow that IBIS generates to depict the action of pushing the button. To refer to the clear button itself, the Sentence Generator uses a definite noun phrase, whereas IBIS highlights the object in the picture.

A mechanism for communication.

Since both generators understand the same formalism, they can provide more information to each other about the resources they have selected simply by annotating the content description. Thus, the content description serves as a blackboard to which all processes can write messages. We use this facility for coordinating the internal text structure with pictures.

Bidirectional interaction

Certain types of coordination between media can only be provided by incorporating interacting constraints between text and graphics. Two-way communication between the media-specific generators may be required as they carry out their individual realizations. Furthermore, coordination may only be possible once partial decisions have been made by the media-specific generators. For example, the text generator needs

to know how the graphics generator has depicted an object before it can refer to the object's visual properties in the illustration. Here we discuss two types of coordination that require bidirectional interaction: coordination of sentence breaks with picture breaks, and cross-referencing text and graphics.

Coordinating sentence breaks with picture breaks.

In addition to revealing several dimensions along which to assign information to media, our media coordination experiments also demonstrated a strong preference for tight structural coordination between text and graphics. Our subjects much preferred sentence breaks that coincided with picture breaks. While multiple sentences accompanying one picture were found satisfactory, subjects strongly objected to a single sentence that ran across several pictures.

Including this type of coordination in COMET requires two-way interaction between text and graphics. Both text and graphics have hard and fast constraints that must be taken into account to achieve sentence-picture coordination. IBIS uses a variety of constraints to determine picture size and composition, including how much information can easily fit into one picture, the size of the objects being represented, and the position of the objects and their relationship to each other. Some of these constraints cannot be overridden. For example, if too many objects are depicted in one picture, individual objects may be too small for clarity.

This situation suggests that constraints from graphics be used to determine sentence size and thereby achieve coordination between picture and sentence breaks. However, some grammatical constraints on sentence size cannot be overridden without creating ungrammatical — or at least very awkward — text. Each verb takes a required set of inherent roles. For example, "put" takes a medium and to-location. Thus, "John put." and "John put the book." are both ungrammatical. Once a verb is selected for a sentence, this can in turn constrain minimal picture size; the LF portion containing information for all required verb roles should not be split across two pictures. Consequently, constraints from text must also be taken into account.

In COMET we incorporate this interaction by maintaining two separate tasks that run independently, each annotat-

ing its own copy of the LF when a decision is made and querying the other when a choice about sentence or picture structure must be made. Once a verb is selected for a sentence, the text generator annotates its copy of the LF by noting the roles that must be included to make a complete sentence. At the same time, the graphics generator annotates its LF with the mapping from the pieces of information to be communicated by graphics to the identifiers of the illustrations in which it intends to communicate the information.

When different sentence structures are possible, the text generator uses the graphics generator's annotations to make a choice by unifying the graphics generator's LF with its own. Consider the example of clearing the display shown in Figure 1. IBIS generates one picture showing the action and its effect; the text generator produces one sentence incorporating the effect as a purpose role of the action ("... to clear the display"). IBIS could depict this action in many ways, depending on the situation. For example, a pair of "before" and "after" pictures can be used, the first showing the action "push" about to occur and the second showing the cleared display. This picture structure can be especially useful in locating the objects participating in the action by showing their appearance prior to this event. Figure 5 (on the next page) shows what happens when IBIS's style rule base is modified so that a composite "before" and "after" pair is preferred. After consulting the annotated logical form, the text generator produces two separate sentences.

Cross-references. One important goal of media coordination is to allow material in one medium to cross-reference material in another. COMET provides for two kinds of cross-referencing: structural and content. The former refers to the coarse structure of the material being referenced. For example, a sentence could refer to an action by mentioning that it is depicted in one of the two pictures on the display. In contrast, a content cross-reference refers to the material's content, such as an object's position in a picture or the way in which the object is highlighted. Structural cross-references require only high-level knowledge of how the material being referenced is structured, whereas content cross-references require low-level

Clear display



Press the CLR button.
This will cause the display to clear.

Figure 5. An alternative explanation for clearing the radio display. A composite illustration containing two pictures is generated with coordinated text (compare with Figure 1).

knowledge of how the material's communicative goals are realized.

COMET's text generator can make both structural and content cross-references to IBIS illustrations. IBIS constructs a representation of each illustration it generates that is indexed by the LF. This representation contains information about the illustration's hierarchical structure, the identity and position of the objects that it depicts, and the kinds of illustrative effects used in constructing the illustration (like highlighting or cut-away views). The text generator queries this representation to generate cross-references. For example, it can refer to information that is communicated "... in the left picture" (structural cross-reference) or mention "... the old holding battery shown in the cut-away view" (content cross-reference).

Current status and limitations

COMET can provide instructions for maintenance and repair procedures in two different contexts. A user can directly request instructions for a specific

procedure through a menu interface (essentially, asking "How do I do x?"). Alternatively, during symptom diagnosis, the user can request an explanation for any diagnostic procedure the system specifies must be carried out. COMET can explain over 40 complex procedures represented in the knowledge base. A variety of explanations are provided for a single procedure, depending on user background, explanation context, or previous discourse. For example, COMET can vary the vocabulary (like not using the word "polarity" if it isn't in the user's vocabulary), the illustration design (like attempting to reuse the preceding illustration's viewing specification for the next illustration to avoid confusing camera motion), or the information communicated (like omitting an explanation for a procedural step if it was explained recently). COMET also has preliminary facilities for answering follow-up questions of the form: "What is an x?", "Where is the x?", or "Why should I do x?"

COMET was designed to be as domain independent as possible. It can be adapted with minimum effort to a new task-based domain when tasks and actions are encoded using a standard plan-based representation¹¹ and objects are

represented using frames. The content planner can produce the content for explanations of tasks or actions represented as plans, as well as for follow-up questions. Because many artificial intelligence systems use plans and frame-based representations, COMET's explanation facilities can be used in a wide range of applications. Similarly, the Sentence Generator grammar, the Lexical Chooser rules, and IBIS rules apply to any domain, task-based or not. In fact, these components have already been used in a number of applications under development at Columbia University. The media coordination rules we developed also apply to any domain that includes actions, physical objects, and abstract relations.

To handle a new domain, it would be necessary to augment the lexicon (adding new vocabulary), the Loom knowledge base (adding new plans and objects), and the graphics knowledge base (adding the new objects' detailed geometry and physical properties). These are currently substantial tasks, but so is the effort required to create conventionally authored explanations for a new domain. Also, note that some domain-dependent information, such as the graphics knowledge base, would ideally be available in CAD/CAM databases created when the objects to be documented were designed. COMET's rule bases were designed for domains involving physical objects. Therefore, handling domains that stress abstract relations among abstract concepts (like statistical analyses of numeric variables) would require major changes. The changes include different content-planning strategies, a different media coordinator grammar, and new graphics generation approaches (for example, adding a method for designing quantitative data displays^{9,12}).

The COMET testbed has allowed us to explore many ways to coordinate the generation of text and graphics. Our present and future work on COMET and its components includes the development of additional generators that support the temporal media of speech and animation. (IBIS already allows for direct, dynamic user control of the viewings specification.⁷) Our work also includes the design of a browsing/navigation facility

for COMET's explanations. We are developing a media layout component that will rely on the media generators' annotations to determine the relationships among pieces of text and graphics. This will allow COMET to group related items spatially.

We plan to allow feedback from the media generators to affect assignments made by the media coordinator and the selection of communicative goals made by the content planner. Our use of unification has the potential to make such feedback possible. Currently, we use an overall control structure for efficiency, calling the unifier separately for each grammar. Instead, we could call the unifier once for the combined series of grammars, thus allowing complete interaction through unification among the types of constraints. In this scenario, a decision made at a later stage of processing can propagate back to undo an earlier one. For example, information about the syntactic form selected can propagate back to the lexical chooser to influence verb choice.

While COMET is a research prototype, we believe that far more powerful systems will someday generate high-quality multimedia explanations for users in a variety of domains. Potential applications include education (explaining scientific phenomena) or even basic home repairs (coaching the user through troubleshooting a broken appliance). Although the hardware needed to run our current system is beyond the reach of most users, rapid improvements in the price-performance ratio will soon make high-performance real-time 3D graphics a fundamental capability of any computer system. In our work on COMET, we have attempted to lay some of the groundwork for the kinds of knowledge-based user interfaces that technological advances will soon make feasible. ■

Acknowledgments

Research on COMET is partly supported by Defense Advanced Research Projects Agency Contract N00039-84-C-0165, the Hewlett-Packard Company AI University Grants Program, National Science Foundation Grant IRT-84-51438, New York State Center for Advanced Technology Contract NYSSTF-CAT(88)-5, and Office of Naval Research Contracts N00014-82-K-0256, N00014-89-J-1782, and N00014-91-J-1872. COMET's development is an ongoing group

effort and has benefited from the contributions of Michael Elhadad (FUF), Dorée Seligmann (IBIS), Andrea Danyluk (diagnostic rule base), Yukiko Fukumoto (media coordinator), Jong Lim (static knowledge base and content planner), Christine Lombardi (media coordinator), Jacques Robin (Lexical Chooser), Michael Tanenblatt (knowledge base), Michelle Baker, Cliff Beshers, David Fox, Laura Gabbe, Frank Smadja, and Tony Weida.

References

1. S. Feiner and K. McKeown, "Coordinating Text and Graphics in Explanation Generation," *Proc. Eighth Nat'l Conf. Artificial Intelligence*, AAAI Press/MIT Press, Menlo Park, Calif., 1990, pp. 442-449.
2. K.R. McKeown et al., "Language Generation in COMET," in *Current Research in Language Generation*, C. Mellish, R. Dale, and M. Zock, eds., Academic Press, London, 1990.
3. R. MacGregor and R. Bates, "The Loom Knowledge Representation Language," Tech. Report ISI/RS-87-188, Information Sciences Institute, University of Southern California, 1987.
4. J. Allen, *Natural Language Understanding*, Benjamin Cummings Publishing Company Inc., Menlo Park, Calif., 1987.
5. R. Reddy et al., "The Hearsay Speech Understanding System: An Example of the Recognition Process," *Proc. IJCAI 73*, Morgan Kaufmann Publishers, Palo Alto, Calif., 1973, pp. 185-193.
6. M. Kay, "Functional Grammar," *Proc. Fifth Mtg. Berkeley Linguistics Soc.*, Berkeley Linguistics Society, Berkeley, Calif., 1979.
7. D. Seligmann and S. Feiner, "Automated Generation of Intent-Based 3D Illustrations," *Computer Graphics (Proc. SIGGRAPH 91)*, Vol. 25, No. 4, July 1991, pp. 123-132.
8. Y. Arens, L. Miller, and N. Sondheimer, "Presentation Design Using an Integrated Knowledge Base," in *Intelligent User Interfaces*, J. Sullivan and S. Tyler, eds., Addison-Wesley, Reading, Mass., 1991, pp. 241-258.
9. S. Roth, J. Mattis, and X. Mesnard, "Graphics and Natural Language as Components of Automatic Explanation," in *Intelligent User Interfaces*, J. Sullivan and S. Tyler, eds., Addison-Wesley, Reading, Mass., 1991, pp. 207-239.
10. W. Wahlster et al., "Designing Illustrated Texts: How Language Production is Influenced by Graphics Generation," *Proc. European Chapter Assoc. Computational Linguistics*, 1991, pp. 8-14.

11. E. Sacerdoti, *A Structure for Plans and Behavior*, American Elsevier, New York, 1977.
12. J. Mackinlay, "Automating the Design of Graphical Presentations of Relational Information," *ACM Trans. Graphics*, Vol. 5, No. 2, Apr. 1986, pp. 110-141.



Steven K. Feiner is an associate professor of computer science at Columbia University. His research interests include image synthesis, applications of artificial intelligence to computer graphics, user interfaces, animation, hypermedia, virtual worlds, and visualization.

Feiner received a PhD in computer science from Brown University in 1987. He serves on the editorial boards of *Electronic Publishing* and *ACM Transactions on Information Systems*, and has coauthored the book *Computer Graphics: Principles and Practice*, 2nd ed. Feiner received an Office of Naval Research Young Investigator Award in 1991 and is a member of the IEEE Computer Society and ACM.



Kathleen R. McKeown is an associate professor of computer science at Columbia University. Her research interests include natural language generation, user modeling, expert system explanation, and natural language interfaces.

McKeown received a PhD in computer science from the University of Pennsylvania in 1982. She was elected president of the Association of Computational Linguistics for a term beginning in January, 1992, and currently serves as its vice president. McKeown has served on the executive council of the Association for Artificial Intelligence and was program cochair of its annual conference in 1991. She received a National Science Foundation Presidential Young Investigator Award in 1985.

Readers may contact Steven K. Feiner and Kathleen R. McKeown at the Department of Computer Science, Columbia University, 500 W. 120 St., New York, NY 10027.

Spatio-Temporal Composition of Distributed Multimedia Objects for Value-Added Networks

Thomas D.C. Little, Boston University

Arif Ghafoor, Purdue University

Future multimedia information systems will require the integration of various objects retrieved from databases distributed across the network.

Here we discuss a methodology for spatial and temporal composition of multimedia objects.

Multimedia is an emerging application-oriented technology embracing many computer disciplines, including interface design, networks, and databases. The field presents many interesting challenges for research. Advances in workstation and network technologies — and the desire to improve communications — have created great interest in multimedia applications. These applications require the processing, storage, and transmission of such data types as audio, video, text, graphics, and images.

One of the most important requirements of a distributed multimedia information system (DMIS) is the integration, or *composition*, of multimedia objects retrieved from databases distributed across a network. We define *multimedia objects* to be aggregate units composed of data elements taken from the aforementioned data classes. Their composition depends on the spatial and temporal characteristics of the multimedia elements.

Spatial composition of multimedia data is unique to each medium. It describes the assembly of objects in space on a workstation display at certain points in time. For pictorial representations like still images and graphics, composition operations include overlay and mosaic, and require processing such as scaling, cropping, color conversion, and position registration. For audio data, composition consists of mixing signals. Other "spatial" audio operations include gain and tone adjustments, which are useful in videoconferencing applications to prioritize one speaker's voice among many. Gain or tone differences can signify "distance" in participants via signal-distortion techniques.¹ A typical spatial composition for pictorial data is shown in Figure 1, which was constructed using scaling and mosaic-forming operations.

Temporal composition requires evaluating the relationships among component elements and scheduling their retrieval and presentation accordingly. The relationships can occur naturally, as for live audio and video, and they can also

be synthetic, consisting of arbitrary temporal constraints on any multimedia data type.² Figure 2 shows various data elements retrieved from storage and presented serially at the times indicated.

The primary issue we address is the overall process necessary to perform spatial and temporal data composition for a DMIS. With respect to delays introduced through the network, we find that temporal composition can be most suitably achieved at the workstation. Spatial composition is most effectively performed in a hierarchical fashion as dictated by the availability of system resources. The subsequent composition methodology combines spatial and temporal composition as a network service.

We also investigate database organizations and data distributions, and discuss spatial and temporal composition functions and their composition into the network architecture. Finally, we address the issue of mapping the composition process onto the network resources as a value-added service.

Databases and distributed objects

A number of common database-management systems (DBMSs) can be identified for multimedia applications. These systems can be organized as centralized, master-slave, and federated, as shown in Figure 3. In the simplest case, a multimedia information system resides on a single server system incorporating DBMS functionality for each medium. Data composition occurs entirely at the server. However, access to remote databases expands the potential applications for multimedia services, requiring a multiple database organization as in Figure 3b and c. Here, we define a data *server* as an intelligent database machine that can perform complex multimedia database operations such as browsing, linking, and pattern-matching. The server can also perform standard DBMS operations such as selection, projection, and join.

The most suitable database organizations for large-scale multimedia applications are the federated and master-slave types as shown in Figure 3b and c. In either case, users must be able to query across the universe of avail-

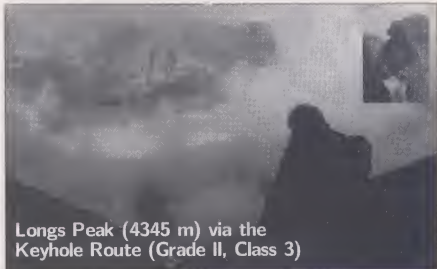


Figure 1. Cropping and overlaying images, and text formatting.

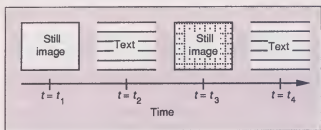


Figure 2. Temporal composition.

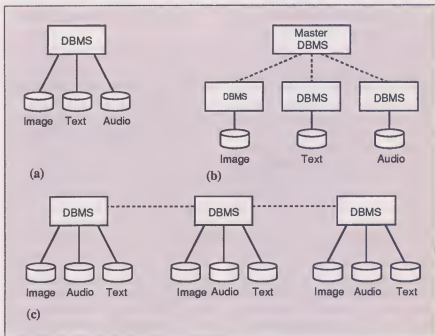


Figure 3. Centralized (a), master-slave (b), and federated (c) database-management systems.

able data. This requires global naming of data and resolution of heterogeneity between DBMSs, server hardware, data formats, etc. After data elements are identified and located, the spatial and

temporal composition of accessed data can be viewed as an added service for the creation of complex multimedia objects. The remaining tasks include specifying the composition function and dis-

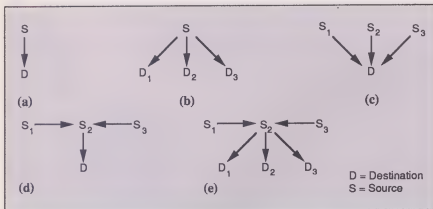


Figure 4. Multimedia connections in a distributed information system: single source to single destination (a), single source to multiple destinations (b), multiple sources to single destination (c), multiple sources to single destination using an intermediate site (d), and multiple sources to multiple destinations using an intermediate site (e).

tributing it over the set of computing elements in the DMIS in a manner appropriate for the DBMS.

In a centralized environment where the data are not distributed (Figure 3a), a single server performs all object composition. However, the centralized environment is not rich enough to support most multimedia applications, since only local data are accessible. For a DMIS, multiple database servers can perform spatial composition to reduce the load on the destination workstation and the required network bandwidth for object communication. In particular, spatial operations such as generating mosaics, cropping, and color conversions, which are common in window-based environments, can significantly reduce superfluous data transfer when performed at server sites. Temporal composition does not exhibit the same benefit at remote servers because no data reduction occurs between source and destination. For strictly temporal composition without spatial composition requirements, a point-to-point virtual connection between source and destination best maintains synchronization for continuous communication and presentation of objects. No additional processing can occur within the network at some intermediate site.

Based on the assumption of a distributed DBMS organization, five types of point-to-point connections for object retrieval and composition in a DMIS can be identified. These composition architectures, shown in Figure 4, include

- single source to single destination,
- single source to multiple destinations,
- multiple sources to single destination,
- multiple sources to single destination using an intermediate site, and
- multiple sources to multiple destinations using an intermediate site.

Figure 4a illustrates a point-to-point connection for a client-server relationship between a single multimedia server and the workstation. Figure 4b represents a shared-object architecture in which a single object is displayed simultaneously to various users via multicasting. This mode enables computer-supported cooperative work (CSCW),³ or teleconferencing, and can require concurrency and consistency control mechanisms for shared objects. Figure 4c represents a distributed-object environment, for which complete composition is performed at the sink in a multidrop fashion. This case can be handled by independent network connections between the sink and each source. However, it poses a challenge to workstation control of intermedia synchronization.

Figure 4d shows a scenario in which objects are composed at an intermediate site after arriving from distributed sources and are sent via a single connection to the final destination. By using a single connection, data sequencing ensures strict ordering and thereby provides intermedia synchronization.⁴

Figure 4e illustrates the general multicast and multidrop case of distributed

object composition in a shared-object environment.

As mentioned, the process of data composition can be distributed within the network to minimize various system costs. In particular, the criterion for the selection of composition locus is a function of various system costs including communication, storage, processing, and desired system-performance characteristics such as the reliability and quality of communications. The problem is analogous to optimization of queries in a distributed DBMS.⁵ However, some differences exist, which we discuss later.

Spatial composition

Current data-composition schemes incorporating video data are mostly analog. For example, image mosaics are generated by capturing and placing video stills from the analog output of a videodisc player. Analog schemes differ from the composition of video, images, audio, and text in a completely digital domain. In the digital realm, we can use storage, processing and communication technology, and open system applicability to achieve device independence in an integrated environment.

The task of composing data spatially as shown in Figure 1 can be specified using an object-oriented paradigm and implemented using DMIS components. Many data representations for composite multimedia data have been proposed based on the object-oriented paradigm.⁶ We concentrate on the task of composition rather than discussing a complete object-oriented spatial data representation. We assume a simple layout structure based on ODA⁷ (Office Document Architecture), as shown in Figure 5. Figure 5a shows a typical spatial translation. Figure 5b illustrates a spatial data representation, or a spatial hierarchy for these translations.

If data are composed at the workstation, user interaction can be most readily incorporated into the composition process. For example, users can choose window placement, resolve occlusion, evaluate view-space coordinates, and perform scaling without dialogue with a remote composition service. Similarly, object modification is most readily achieved on the workstation. However, some objects can exceed the local storage capacity of the workstation, and modification must occur on frag-

ments of the original object stored remotely.

Composition at the workstation requires transmission of complete multimedia objects from the database, such as full-size color video or full-resolution images. At the destination, much of this resolution is not used after scaling, cropping, color mapping, or other spatial operations. A remote service, by contrast, can perform these spatial operations prior to data transmission, reducing the transmission overhead and freeing the workstation for other tasks. Specialized hardware like array processors and video special-effects devices optimized for high-speed spatial manipulation can also reside at the composition sites.

To provide a value-added service facilitating these kinds of spatial operations within a network, we describe a set of spatial manipulation primitives that can be implemented as standard functions at various sites within the DMIS. These primitives fall into two classes: unary and binary operations. Unary operations are applied to single data objects like the low-pass filtering of an audio segment or image cropping. Binary operations combine pairs of objects, creating their composition. Table 1 summarizes examples of these operations and lists approximate processing costs.

We envision spatial composition to be a multiphase process consisting of unary operations that adjust the data elements, followed by binary operations that compose the adjusted elements into final form. Further, we define a logical entity, called the *spatial translator* (ST), which can be distributed throughout the network to perform these operations. Note that these operations constitute the overall translation from storage to workstation display, a simplification of the many intermediate transformations between the various coordinate systems and canonical representations usually indicated in graphics programming standards like the Graphical Kernel System (GKS) and Programmer's Hierarchical Interactive Graphics System (PHIGS).

Distributing operations with a uniform interface throughout the DMIS lets users tailor spatial data composition to individual object requirements. As data are retrieved, users can invoke spatial operations at remote servers through the uniform interface to reduce data traffic.

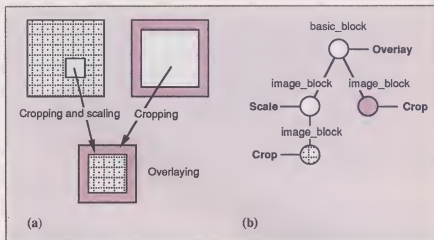


Figure 5. Multiphase spatial composition: transformations (a) and spatial hierarchy (b).

Table 1. Characteristics of typical spatial manipulation primitives.

Operation	Operands	New Size	Cost Estimate	Data Types
Overlay (mix)	A, B	$\max(A , B)$	$\min(A , B)zP_m$	Image, video, graphics, and
Overlap	A, B	$>\max(A , B)$	$< A + B $	
Abut (concatenate)	A, B	$ A + B $	$(A + B)zP_m$	
Crop (cut)	A	$c A $	$c A zP_m$	audio
Scale (filter)	A	$s A $	$ A P_s$	Text
Text format	A	$f A $	$f A zP_m$	

$|A|$ = Size of A in image pixels, audio samples, or text characters

c = Fraction of unaffected pixels in a crop operation

f = Number of pixels per formatted character

P_m = Unit cost per memory move

P_s = Unit cost per scaled pixel

s = Scale factor

z = Number of words per image pixel, audio sample, or text character

As mentioned, spatial operations are basically transformations on multimedia objects to achieve such functionality as cropping, filtering, and overlaying. However, some operations may result in data reduction, while in other cases the new data elements can be larger than the old (for example, coloration and scale-up). If the criteria to select between a server or workstation for composition are based solely on reduction of network traffic, clearly we want to perform unary spatial transformations at the server site only when a data reduction occurs. However, if the criteria

include minimizing processing at the workstation, we may want to perform the data-increasing transformations at the server as well.

Considering binary spatial manipulation, suppose the criterion to select a site for composition is based on reducing network traffic and minimizing the workstation processing load. Let R_1 and R_2 represent two objects to be merged spatially in some arbitrary way, with characteristic data sizes of $|R_1|$ and $|R_2|$. The binary spatial merge for some transformation

b_T

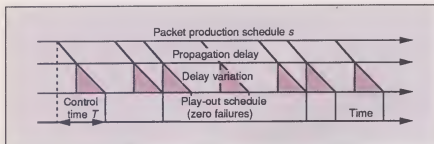


Figure 6. Playout in the presence of random network delays.

will result in a final display object R_d with size $|R_d|$ and is denoted as

$$R_d = b_s(R_1, R_2).$$

Clearly $\max(|R_1|, |R_2|) \leq |R_d| \leq |R_1| + |R_2|$, since the merge represents in the worst case the union of the two objects (for instance, overlay, abut, and mix). This relationship is valid for any type of data object including text, image, video, and audio, assuming the absence of data compression.

If $|R_d| = \max(|R_1|, |R_2|)$, performing composition at the server potentially reduces data traffic to the workstation by $\min(|R_1|, |R_2|)$. On the other hand, if $|R_d| = |R_1| + |R_2|$, no savings result. Within this range, the choice of a site for composition depends on the penalty associated with the composition processing versus the necessary data communication bandwidth and the composition requirements of other related objects. The overall savings in data transmission includes all operations required in forming the final composite object.

For example, consider again the multimedia object in Figure 1 (shown here in gray scale). The data constituting the object are assumed to exist in several distributed servers. Each image is stored in 24-bit, full-color format (8 bits per color, three colors per pixel). The sub-image in the figure is created by cropping its corresponding (1,200 × 925-pixel) source image to a size of 80 × 120 pixels. This is superimposed onto the background image along with the ASCII text (61 characters × 8 bits per character) with a selected font. The final composite object is entirely in bitmapped form and is the size of the cropped background image (888 × 685 pixels).

If composition is performed at the workstation, the raw, unprocessed images and text would have to be transmit-

ted to that location (1,200 × 925 pixels × 24 bits/pixel × 2 images + 61 characters × 8 bits/character = 53,280,488 bits). If composition occurs before transmission, the data-transmission requirement is 888 × 6 pixels × 24 bits/pixel = 14,598,720 bits. The savings in data transmission for remote composition is 38,681,768 bits, or a traffic reduction of 73 percent.

Note that in these calculations we have assumed no data compression or the overheads associated with storage and communication. However, the percentage savings remains unchanged with compression. This example illustrates the reduction of data possible during the process of data composition at some intermediate site in the network as data are retrieved from a database, merged, and sent to the final presentation site.

The spatial processing requirements depend on the size and type of the object considered, the type of composition function, and its implementation, whether parallel or serial. Table 1 summarizes processing approximations for some of the various spatial operations on images, audio, and text. Either time or number of operations can be applied as costs. For example, the move cost relates to memory-access time; that is, P_m equals two memory-access cycles (one read, one write).

For a given composite multimedia object, the spatial formatting requirements can be quantified by using the cost estimates for the various spatial transformations. Chaining the cost estimates as described by the spatial hierarchy provides an estimate of overall processing cost. The evaluation of such processing requirements is important in determining the real-time computational performance of the workstation and the distribution of composition operations within the network.

Temporal composition

Multimedia data objects are time-dependent and must be synchronized accordingly. This synchronization requirement encompasses static objects, such as still images and text, and continuous streams of audio and video. A DMS must satisfy this requirement in the presence of random network delays due to the inherently asynchronous nature of a packet network^{7,8,9} and the latencies associated with storage devices. This problem is particularly acute, since several streams of different origin can require synchronization with each other. For continuous streams of data, the problem is to ensure the proper time of play-out (presentation time) of each data element in spite of random network delays, as illustrated in Figure 6.

Synchronization of this type has been typically applied to single streams of packetized audio and video data⁸ but can be generalized to multiple streams and nonstream data such as still images and text. Approaches to synchronization determine the delay and buffering required to establish a level of packet loss for various network delay distributions. This delay, called control time T , can be found for audio and video streams⁹ given a target packet-loss probability. The same principle can be applied to nonstream data. However, in the former analyses, it is assumed that the packet-generation rate is equal to the consumption rate and that the capacity of the communication channel is not exceeded. When arbitrary sequences of multimedia objects are presented, the data can easily exceed the channel capacity for some intervals. However, due to flexibility in object retrieval for stored-data applications, data-retrieval times can be reorganized so that the channel capacity does not adversely impact the presentation. Of course, this is not possible for live data sources. In essence, database sources provide more freedom in controlling when data are acquired by the application, as shown in Figure 7.

Beyond the problem of compensating for random network delays, multiple-stream synchronization can be controlled by the destination workstation or by any other intermediate server within the network prior to data delivery. An object's temporal specification needs

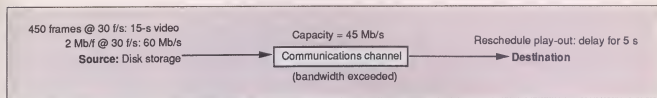


Figure 7. Flexibility in database retrieval.

to be known to the synchronization controller irrespective of the controller location within the network. A Petri-net-based approach to specify the temporal requirements for multimedia objects can be used for this purpose.^{1,10} If synchronization is performed at the destination (as in Figure 4c), the workstation must evaluate the temporal specification of objects and carry out a synchronization dialogue with the remote servers prior to data transfer. If synchronization occurs at any other intermediate server (as in Figure 4d), the workstation does not need to evaluate the temporal specification. However, due to such factors as network latencies, retransmission (if applicable), and workstation-performance limitations, significant skew can be introduced among these synchronized streams, and an intermediate server has little control over the final result at the workstation. Therefore, an intermediate server cannot reliably provide the fine synchronization required for audio/video streams, which generally require a skew of less than 150 milliseconds.

A typical application might use the synchronization service for preorchestrated presentations, teleconferencing, or CSCW. Typically, an object is identified from a database through browsing or querying operations. Once identified, the object can be retrieved, composed, and presented. Since both spatial and temporal composition specifications must be met, an important operation in this scheme is the decomposition of multimedia data into classes for independent transfer. This separation can increase network performance by identifying unique data-traffic classes and using different transfer protocols tailored to each class.¹¹ In essence, such transport protocols can provide different levels of guaranteed quality of service for each data type based on its tolerance to packet delay and loss. For example, an image object

requires error-free service, while audio objects can tolerate errors.

We present two communication protocols to perform synchronization as a value-added network service between sources and a destination.¹² The Application Synchronization Protocol (ASP) and Network Synchronization Protocol (NSP) allow the communication of complex multimedia presentations from distributed sources for play-out at a single site. The protocols utilize a Petri-net-based temporal specification of a multimedia object.¹² This model basically specifies the precedence relations among all subobjects in the form of a partial and strict ordering. Partially ordered objects have simultaneous play-out deadlines and require concurrent presentation, while strictly ordered objects require time-sequential play-out (as in Figure 2).

The ASP sets up and initiates data transfer as specified by temporal requirements on an end-to-end basis. Spatial requirements of an object affect the ASP, since they must be passed during connection setup to the remote sites. The ASP takes as input a selected object representing the aggregation of a complex multimedia presentation requiring synchronization. It then returns independent streams of synchronized data traffic that can then be routed to specific output devices for play-out at the workstation. The ASP also provides control over the quality and the cost of transmission service by negotiating a target packet-loss probability and delay with the underlying guaranteed-service data-transport mechanism.¹¹

The NSP provides a data-transfer facility with a predicted end-to-end delay characteristic based on the specified probability of late packets. Neither the ASP nor the NSP specifically provides a mechanism for reconstructing late or lost packets. They rely on selecting the appropriate quality of service parameters

for each medium's tolerance to delay and loss.

These protocols allow synchronization of independent network connections, unlike approaches to synchronization¹ that require sequencing of synchronized data onto a single virtual circuit (as in Figure 4c). For the ASP, it is assumed that all synchronization is performed at the destination. Intermediate nodes are considered only if some intermediate spatial composition function is needed. Otherwise a point-to-point connection is implied with corresponding end-to-end properties.

In summary, the ASP and NSP provide synchronization, which involves the following steps:

- (1) retrieving the spatio-temporal relationships that describe the components constituting the complex multimedia object.
- (2) evaluating the precedence relationships in the Petri net, thereby creating a play-out schedule.
- (3) decomposing the schedule into subschedules based upon the different traffic classes represented and the locations of stored data.
- (4) determining the overall control time required to maintain synchronization among the traffic classes through interaction with the NSP and cooperating sites.
- (5) providing synchronous data transfer.

Spatio-temporal composition

Here we investigate a combined approach to performing temporal and spatial composition of a DMIS as a service provided by the network. We envision the heterogeneity of the network in terms of speed, computational resources, and topology to motivate the use of a hierarchical composition process. Multiple

data servers (DSs) and intermediate sites, or composition servers (CSs), collaborate to compose the requested objects both spatially and temporally.

Generally, an object model consists of information describing the various operations necessary for spatial-temporal composition. The model is stored in the network at a central site. At the time a session is established by a user, the information is identified from the central site, and the object hierarchy is decomposed and mapped onto the set of servers (DSs, CSs, and the workstation).

Although the problem of assignment of composition locus for a given multimedia object is analogous to query optimization for distributed databases,² it differs in several ways. Because a sequence of spatial operations often cannot be permuted, little optimization can be achieved through reorganizing the sequence of spatial operations. Some spatial transformations (like scale-up) increase data volume and are optimal when performed closer to the data's destination rather than near its source. Second, the optimization technique assumes homogeneity in processing and communication costs and therefore does not account for specialized hardware for each medium, nor does it consider the necessity for load balancing when long-lived database transactions are

present (as in the retrieval and play-out of movies).

As mentioned, temporal composition always requires control by the workstation, since independent media cannot

be combined in any reasonable manner at intermediate sites in the network en route to the play-out destination, except to provide sequencing.⁴ On the other hand, spatial composition can be performed at either the composition servers or at the destination workstation. The choice depends on the object characteristics, the workstation storage and computational capability, and the bandwidth of the network. If remote composition is dictated, the composition is delegated to at least one CS, which we call the primary CS. The secondary servers consist of data sources. The choice of primary server should allow for

- the locality of objects,
- the amount of spatial processing required on the selected objects,
- the spatial composition capability of each CS (for example, an array processor), and
- the current CS loading.

The first factor means that the composition server with the closest proximity to the largest percentage of data is optimal.³ The spatial processing consideration should be weighed in determining the primary server. By analyzing the spatial organization of a complex object, one can map a composition function to a specific CS or set of CSs, based upon CS requirements and the capabilities of the

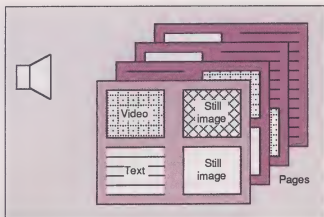


Figure 8. Pages of the electronic magazine.

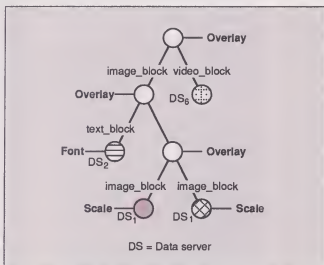


Figure 9. Spatial hierarchy for a page.

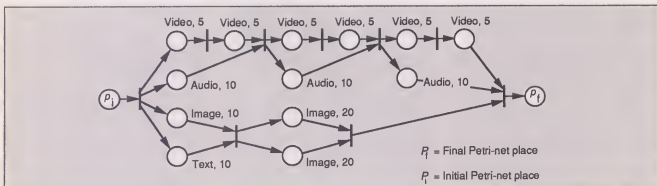


Figure 10. Temporal specification for a page.

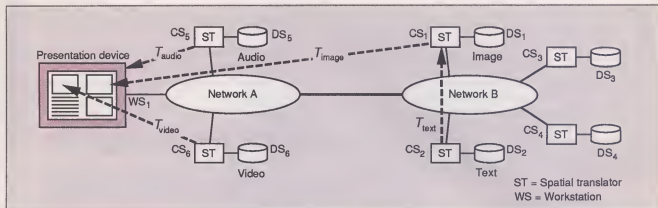


Figure 11. Network example.

servers. Also, the load on a CS can affect the primary server selection.

With respect to the workstation and network components, the assignment of spatial operations must allow for the optimal utilization of bandwidth. This can dictate reducing objects at the CS (through scaling, cropping, and filtering), enlarging objects at the workstation (scaling, coloring, and formatting text), and building windows at the workstation (dealing with occlusion). To minimize workstation computational requirements, all spatial operations must be performed at the CS, including the window management.

Clearly there is a trade-off between traffic on the network, computational load at the workstation, and user control over presentation. When objects are composed at the CSs, the user cannot control data assembly, since this operation is performed prior to the reception of the data. A compromise can be achieved by specifying the object hierarchy so that some objects must be assembled at the workstation and others can be distributed to various CSs. Based on these considerations, we present a scenario for combined spatial and temporal object composition using the ASP and ST entities. We chose the electronic magazine as an example.

This multimedia application is analogous to a printed magazine. A "reader" may browse through "pages" to read articles and view pictures and audio/video presentations, as shown in Figure 8. We model the pages of this application as objects that are browsed sequentially. In addition, the user may perform queries or searches to locate specific articles or advertisements. Spatial and temporal composition is required for

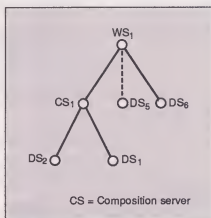


Figure 12. Mapping of spatial operations to composition servers.

elements of text, image, audio, and video, within the context of a page as indicated by the spatial and temporal specifications of Figures 9 and 10. These data are assumed to be distributed across a high-speed network and require remote data access and composition.

After a page is selected for presentation at the workstation, a CS is identified by consulting some centralized *name server* that maintains a global table of objects, names, locations, and other characteristics. The server also maintains DMIS global state information regarding the availability of resources and the load at each CS. This information is applied in the selection of a primary CS for each class of data used to construct an object. Subsequently, the workstation uses the object's temporal specification (Figure 10) to establish an ASP session. The ASP then initiates individual concurrent connections to the

primary CSs using the NSP on a point-to-point basis. The primary CSs establish NSP connections with DSs as required for the indicated spatial composition. Load is effectively balanced by placing new sessions onto CSs with the least load, if appropriate, in light of other primary server selection criteria. Based on the DMIS architecture shown in Figure 11, a mapping of spatial operations onto the set of CSs appears in Figure 12. In general, a mapping requires optimization of some cost function such as communication bandwidth.

A synopsis of the evaluation of the temporal requirements by the ASP and NSP is as follows: The temporal specification is interpreted to generate a set of deadlines for each class of multimedia data.¹² Assuming video is synchronized per frame (30 f/s), audio per 10-second interval, and text and image per object, the temporal specification (see Figure 10) generates the subschedules: $S_{\text{video}} = (0, 0.033, 0.066, 0.099, \dots, 29.966)$ s, $S_{\text{audio}} = (0, 10, 20)$ s, $S_{\text{text}} = (0, 10)$ s, and $S_{\text{image}} = (0, 10)$ s. For each subschedule, an NSP connection establishment procedure is invoked, resulting in a set of control times: $T_{\text{audio}}, T_{\text{image}}, T_{\text{text}},$ and T_{video} , which represent the aggregate end-to-end delays over virtual channels between data source and destination for each medium (see dashed lines in Figure 11). The maximum control time T_o is found and returned to the ASP for generation of the overall start time. Data transfer is provided by the underlying communication mechanism that interprets the derived schedule, which is assured to be feasible by the evaluation of the connection establishment procedure of the ASP and NSP.

The composition of data objects in a DMIS is an important technical problem considering the complexity of supporting time-dependent media and data heterogeneity in an open systems environment. Object composition requires consideration of both the temporal and spatial characteristics of multimedia elements.

There are trade-offs to partitioning the object-composition process and its mapping onto the resources of the network based on the communication and computational requirements of an object. The mapping of this process, which combines spatial and temporal composition, results in a value-added service provided by the network. ■

Acknowledgments

We thank the reviewers for their constructive comments. We also thank P. Rivera for providing the photographs that make up Figure 1. Our work was partially supported by the New York State Center for Advanced Technology in Computer Applications and Software Engineering at Syracuse University.

References

1. L.F. Ludwig, N. Pincever, and M. Cohen, "Extending the Notion of a Window System to Audio," *Computer*, Vol. 23, No. 8, Aug. 1990, pp. 66-72.
2. T.D.C. Little and A. Ghafoor, "Network Considerations for Distributed Multimedia Object Composition and Communication," *IEEE Network*, Vol. 4, No. 6, Nov. 1990, pp. 32-49.
3. I. Greif, ed., *Computer Supported Cooperative Work: A Book of Readings*, Morgan Kaufmann, San Mateo, Calif., 1988.
4. W.H. Leung et al., "A Software Architecture for Workstations Supporting Multimedia Conferencing in Packet Switching Networks," *IEEE J. Selected Areas Comm.*, Vol. 8, No. 3, Apr. 1990, pp. 380-390.
5. W.W. Chu and P. Hurley, "Optimal Query Processing for Distributed Database Systems," *IEEE Trans. Computers*, Vol. C-31, No. 9, Sept. 1982, pp. 835-850.
6. D. Woelk, W. Kim, and W. Luther, "An Object-Oriented Approach to Multimedia Databases," *Proc. ACM*, ACM, New York, May 1986, pp. 311-325.
7. *ISO Document No. 8613*, Int'l Organization for Standardization, Geneva, Mar. 1988.
8. M. De Prycker, M. Ryckebusch, and P. Barri, "Terminal Synchronization in Asynchronous Networks," *Proc. IEEE Int'l Conf. Comm.*, IEEE Press, Piscataway, N. J., 1987, pp. 800-807.
9. G. Barberis and D. Pazzaglia, "Analysis and Optimal Design of a Packet-Voice Receiver," *IEEE Trans. Comm.*, Vol. COM-28, No. 2, Feb. 1980, pp. 217-227.
10. P.D. Stotts and R. Furuta, "Petri-Net-Based Hypertext: Document Structure with Browsing Semantics," *ACM Trans. Office Automation Systems*, Vol. 7, No. 1, Jan. 1989, pp. 3-29.
11. A.A. Lazar, A. Temple, and R. Gidron, "Magnet II: A Metropolitan Area Network Based on Asynchronous Time Sharing," *IEEE J. Selected Areas Comm.*, Vol. 8, No. 8, Oct. 1990, pp. 1,582-1,594.
12. T.D.C. Little and A. Ghafoor, "Multimedia Synchronization Protocols for Broadband Integrated Services," to be published in *IEEE J. Selected Areas Comm.*, Dec. 1991.

First International Workshop on Industrial Applications of Fuzzy Control and Intelligent Systems

Keynote Address by Dr. Lotfi Zadeh
Opening Address by Dr. Bart Kosko

November 21-22, 1991, College Station, Texas

The workshop, sponsored by Texas A&M University and the North American Fuzzy Information Processing Society (NAFIPS), brings together world leaders in Fuzzy Control theory and industrial applications in a unique setting. Key industrial participants of the workshop include American Neuralogix, Apronix, Eaton, Hitachi, Honeywell, HyperLogic, Integrated Systems, Matsushita, Mycom, NASA, Omron, Rockwell, Siemens, SGS Thomson, Togal InfraLogic, and Toshiba. To register, fill out the form below and mail it together with registration fee (\$350 before 10/21, \$450 after 10/21) to Walter Daugherty, CFLSR Fuzzy Workshop/Dept. of Computer Science, Texas A&M University, College St., TX 77843-3112. Make check payable to Fuzzy Workshop/Dept. of Computer Science. For hotel reservations call College Station Hilton (409)693-7500 by Nov. 1 (special rates: \$57 sgl, \$63 dbl). For more information contact John Yen (409)845-5466, Fax: (409)847-8578, E-mail: yen@cs.tamu.edu.

Name: _____

Address: _____

Phone: _____

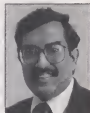
Thomas D.C. Little is an assistant professor in the Department of Electrical, Computer, and Systems Engineering at Boston University. His research interests include distributed systems, computer networking, multimedia database management, and real-time system design.

Little received the BS degree in biomedical engineering from Rensselaer Polytechnic Institute in 1983, and the MS degree in electrical engineering and PhD degree in computer engineering from Syracuse University in 1989 and 1991. He is a member of the IEEE Computer and Communications Societies and the Association for Computer Machinery.



Arif Ghafoor is an associate professor at Purdue University in the School of Electrical Engineering. He also consults with such companies as AT&T Bell Laboratories and General Electric in the area of telecommunications and distributed systems. His research interests include the design and analysis of parallel and distributed systems, and telecommunication.

Ghafoor received his BS degree in electrical engineering from the University of Engineering and Technology, Lahore, Pakistan, in 1976. He received the MS in electrical engineering, the MPhil, and the PhD in electrical engineering from Columbia University in 1977, 1980, and 1985. He is a senior member of the IEEE, and a member of Eta Kappa Nu and the IEEE Computer Society.



Readers may contact Thomas D.C. Little at the Department of Electrical, Computer, and Systems Engineering, Boston University, 44 Cummington St., Boston, MA 02215; e-mail: tdc@buena.bu.edu.

A Continuous Media I/O Server and Its Synchronization Mechanism

David P. Anderson and George Homsy

University of California at Berkeley

Because audio and video are perceived as changing continuously over time, we call them continuous media. Continuous media dominate mass-market information systems such as television, telephone, and recorded music. With advances in conversion and compression hardware, they are becoming viable as computer I/O media as well.

However, CM and computer systems do not coexist easily. Computer hardware (processors, buses, networks) sometimes cannot handle CM data rates. Even if the hardware is sufficient, the software may not provide the needed real-time performance characteristics. Because of these problems, some projects have used architectures that handle CM data in separate systems (analog, digital, or hybrid) operating under computer control. Examples include Vox,¹ Etherphone,² Imal,^{3,4} and Pandora.⁵

In the long run, performance problems will likely be solved, allowing application programs in general-purpose distributed computer systems to handle CM data. We anticipate systems in which

- workstations are equipped with CM I/O devices (cameras, microphones, speakers) that transfer data to and from main memory;
- processor, bus, and memory bandwidth suffice for handling multiple concurrent CM data streams;
- user-level processes, running in protected virtual-address spaces, can read and write multiple streams of CM data;
- communication networks and protocols let processes on different hosts communicate CM data in real time; and
- file servers can store and retrieve CM data in real time.

Typical application programs in such systems might include

- *Multimedia document browser.* This lets the user select portions of a multimedia document, stored on a remote file system, for display. Each portion may

Acme is a network server for digital audio and video I/O. It lets users specify their synchronization requirements through an abstraction called a logical time system.

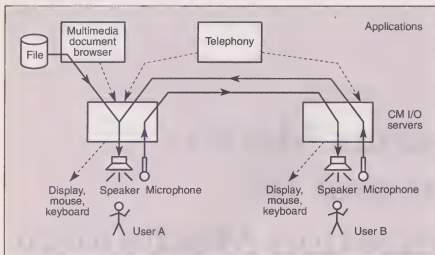


Figure 1. Continuous media I/O servers provide shared, network-transparent access to audio/video I/O at the user interface. In this example, user A is listening to sound from a multimedia document while engaging in a conversation with user B. The application programs (document browser and telephony) can run on machines other than the workstations. Network connections transport CM data (solid lines) and discrete control information (dashed lines).

consist of several parallel streams of audio and video. Graphic controls let the user pause, adjust audio volume, and so on.

- **Telephony and conferencing system.** This lets two or more users hold an audio or audio/video conversation with one another. This facility could be integrated with group-oriented software, perhaps allowing the participants to work together on a document.
- **Multitrack audio/video editor.** This stores audio and video segments on one or more file servers. The segments can be played back, sequenced, and/or overlapped, according to an editable "play list." New material can be recorded and overdubbed.

All these applications need a facility for I/O of CM data to and from a user's workstation. The facility should allow several applications to run concurrently, sharing the workstation's CM I/O devices. Applications should be capable of running on machines other than the workstation. Given the heterogeneity in CM I/O devices and the data types they support, some method of type conversion must be provided to ensure application portability.

These requirements lead to the idea

of a *CM I/O server*. Such a server offers network-transparent access to CM I/O hardware and mediates access between concurrent streams. It is analogous to a network window system, such as X⁶, that provides access to discrete I/O devices such as graphics display, keyboard, and mouse (in fact, there may be advantages in combining the two into a single server).

A CM I/O server is a low-level software layer. Its clients may offer higher level abstractions, such as HyTime documents or MHEG (Multimedia and Hypermedia Information Coding Experts Group) objects,⁷ and interconnection of devices and special-purpose processors.³ A user can run several such applications concurrently, as shown in Figure 1.

CM applications may require that input or output of multiple CM streams start at given relative times and proceed in "lockstep" or that graphics output events occur at given times relative to a CM stream. A CM I/O server must allow clients to express these synchronization requirements, and must also provide a mechanism for enforcing them. The server should allow clients to express the "tightness" of their synchronization requirements, so that the mechanisms (which may be expensive) are used only when needed.

Acme: An I/O server for continuous media

Acme (Abstractions for Continuous Media) is a CM I/O server developed at UC Berkeley. Other experimental CM I/O servers have been described by Terek and Pasquale⁸ and Angebrandt et al.⁹ Acme differs from them largely in its approach to synchronization. Acme provides CM input and output for multiple concurrent clients. It supports several device types: speakers, microphones, video displays, and video cameras. An Acme server controls a fixed set of *physical devices* of these types, and it allows clients to create *logical devices* representing "abstract" versions of the physical devices. Each logical device is bound to a physical device of the same type; multiple logical devices may be bound to a single physical device.

An Acme server communicates over TCP (transmission-control protocol) connections, allowing clients to run on any machine. Control operations, such as the creation of logical devices, are done by remote procedure calls over a *control connection*. CM data streams, both input and output, are conveyed on *data connections*. Data connections may link either a client and a server or two servers. Figure 2 shows the CM data flow within an Acme server.

Physical devices are accessed concurrently; there is no device "locking" or exclusive access. For example, the output from a speaker is the sum of the audio data of the logical devices bound to it; clients specify a volume parameter for each stream. Each logical video display is placed in a video window on the physical display. Data from an input physical device (microphone or video camera) is copied to all logical devices bound to it.

The Acme synchronization abstraction is called a *logical time system*, or LTS. Each logical device is bound to an LTS; an LTS can include several logical devices, of both input and output types. I/O on the logical devices within an LTS is synchronized in the sense that data units with the same time stamp are displayed or collected at about the same real time.

Our implementation of the Acme server runs on the Sun Sparcstation. The server is a user-level program running under SunOS 4.1. It is written in C++ and uses real-time lightweight process-

es (see sidebar titled "The use of processes and objects in Acme"). It uses the Sparstation's built-in hardware for digital audio I/O (8,000 8-bit samples per second), and also provides output of compressed and uncompressed video at 10 frames per second.

Logical time systems: semantics and examples

Why do CM I/O servers need a synchronization mechanism? Imagine a CM I/O server in which data from each stream is queued FIFO and is displayed on its physical device as soon as possible. Suppose an audio and a video stream, each from a different file server, are to

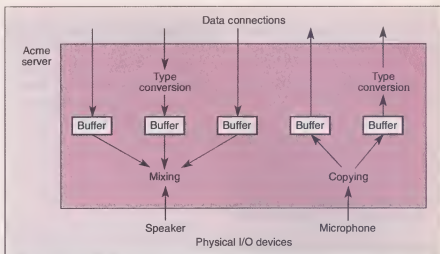


Figure 2. An Acme server handles several CM data streams concurrently. Output is read from a network connection (one per logical device), type-converted if necessary, buffered, mixed, and output to a physical device (a video display or a speaker). Input data is collected from a physical device (a video camera or a microphone), duplicated, buffered, type-converted if necessary, and written to a connection.

The use of processes and objects in Acme

The Acme server is written in C++. It uses abstract base classes extensively, making it easy to add new device types and data formats. Acme structures execution as a set of processes. These processes are "lightweight": They share a single Unix process and address space. Figure A illustrates the three process types described below:

- **Network I/O processes** perform I/O to and from data connections. A network output process reads data from its connection, processes the data if needed (for example, for audio output it converts 8-bit logarithmic audio samples to a linear representation and scales them by a volume parameter), and writes the data to the buffer of the associated logical device. Network input processes do the reverse.

- **Device I/O processes** perform I/O to and from physical devices. These processes are awakened on each I/O completion. A device output process gathers data from the buffers of the associated logical devices, combines them (for example, by summing audio samples), and writes them to the device. A device input process reads data from the device and places a copy in the buffer of each associated logical device.

- **Discrete event processes** handle control operations and events. The ac-

ceptor process accepts new connections, and the reader process reads control requests. Both send messages to a main process that carries out the appropriate action, perhaps sending a reply or event message. Network and device I/O processes may instruct the main process to send an event to a client.

Acme uses a real-time process scheduling policy (earliest deadline first, where deadlines are assigned according to data time stamps). Scheduling is preemptive: For example, if a device I/O process is awakened while a discrete event process is running, the discrete event process will be preempted.

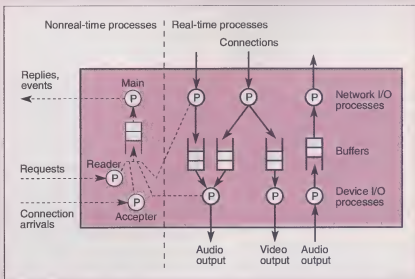


Figure A. The Acme server uses multiple lightweight processes sharing a single Unix address space. Network and device I/O processes handle CM data (solid lines), while nonreal-time processes handle control operations and event reporting (dashed lines).

be played synchronously. Several problems might occur:

- The transmission start times and the network delays may differ. This would skew the display of the data streams.
- Because of a pipelined decompression architecture, the video device might not start displaying until several frames have been received, whereas the audio device starts playing audio samples as soon as they are received. This could also skew the output.
- The actual rates at which each physical device displays data are typically determined by separate quartz crystal oscillators. If the oscillator frequencies do not exactly match, audio and video output will not stay in synchrony over long periods.
- Scarcity of system resources (for example, network bandwidth and CPU time) may cause a stream to "starve." If I/O continues on the other streams, the skew in the output will increase.

LTS operations and parameters. The logical time system abstraction allows Acme clients to specify how to synchronize input and output. To describe LTS semantics, we model CM data streams as a sequence of "units" (video frames or audio samples). Each unit has an associated nonnegative time stamp giving its temporal position relative to the start of the stream. Time stamps may be implicit; in an audio encoding with a constant sampling rate, a given sample's time stamp is determined by its position in the stream.

Each LTS has a *current value* (initially zero) and a group of logical devices. While an LTS is running, its current value increases at approximately the same rate as real time. I/O on each logical device is synchronized to the LTS: A data unit with time stamp t is input or output close to the time when the LTS value is t . The timing resolution is the device interrupt period (tens of milliseconds); unlike the system of Angebrannt et al.,⁹ Acme does not support sample-level synchronization.

A client can create a new LTS using

```
LTS = create_LTS(mode, param);
```

The mode argument specifies how the LTS's rate of advance is determined. Acme provides two options:

The LTS abstraction lets Acme clients specify how to synchronize input and output.

- **Device-driven.** The LTS advances at the rate of a physical device, D_{master} (specified by param). I/O on D_{master} occurs at its natural rate, while other devices in the LTS may require rate adjustment (see below, "Rate matching"). D_{master} is typically the device on which timing errors are most easily perceived (usually, audio rather than video).
- **Connection-driven.** The LTS advances according to the rate of data transfer on a specified connection. I/O is sped up (for example, by dropping output data) if data buffered in the server exceeds a threshold given by param. This is the appropriate choice if low delay is needed.

Other modes are possible. For example, an LTS might be driven by an external timing source, such as a clock, or by the fastest or slowest device in the LTS.

A logical device is bound to an LTS using

```
bind(LTS, logical_device, offset,  
     max_skew, start_count);
```

Data display or collection on the device begins when the LTS value reaches the given offset. Devices can be bound to an already running LTS; in this case, offset should exceed the current LTS value.

If the data transfer rate on a connection is too low (for example, because of network congestion), an output logical device may experience starvation and an input logical device may experience buffer overrun. In such cases we say that the device is *blocked*. If an LTS continues to advance while one of its devices is blocked, the skew between the device and the LTS will increase. The upper bound on skew for this device is specified by *max_skew*. If the skew reaches this value, the LTS stops advancing while the connection catches up.

For output devices, *start_count* specifies how much data must be received on the connection before the LTS can start running. To reduce the likelihood of initial starvation, Acme defers starting an LTS until sufficient data has been received on each output connection. If the network supports connections with bounded jitter,¹⁰ an appropriate value for *start_count* can be calculated from these bounds.

Other LTS operations include:

```
start(LTS);  
stop(LTS);  
time = value(LTS);  
alarm(LTS, time);
```

An LTS is initially in a "stopped" state. After binding all initial devices to it, the client calls *start()*. The *value()* operation returns the current LTS value. The *alarm()* operation requests that the client be sent a message when the LTS reaches the given time value.

Examples of LTS usage. We now illustrate how to use LTSs to express the synchronization requirements of the three example applications introduced earlier:

- **Multimedia document browser.** Suppose a document component involves several audio/video components, each conveyed on a different data connection. Bind the logical devices to a single LTS. Smoothness of display is more important for audio than for video, so use a device-driven LTS, driven by the audio device. Video frames will be repeated or dropped as needed to match the rate of audio output. Since start-up latency is unimportant, a large *start_count* can be used to minimize the chance of starvation. The value of *max_skew* depends on the type of material; if it includes speech, then a small value (100 milliseconds or so) will provide lip-sync.
- **Telephony and conferencing system.** Low end-to-end delay is the main goal here. Bind the logical devices to separate connection-driven LTSs. This prevents buffer buildup if the rate of a microphone at one end exceeds that of the speaker at the other end. Use start counts of zero.
- **Multitrack A/V editor.** When new material is being "overdubbed," the input and output logical devices are bound to a single LTS. Require-

ments are generally the same as for browsing, except that it may be important not to drop any output data. This could be accomplished by using the slowest device to drive the LTS.

Implementing logical time systems

The Acme server arranges to receive a Unix signal for every 256 audio output samples and every 1,024 audio input samples. It also outputs video at 10 frames per second, using a periodic Unix signal. We will refer to these signals as "I/O device interrupts."

The LTS implementation has three components:

- start-up, during which the LTS waits for device and connection pipelines to fill;
- normal operation, during which the LTS handles rate mismatch; and
- starvation/overrun, during which the LTS may temporarily stop.

Start-up synchronization. After the start() operation, the LTS enters a start-up mode. In this mode, data is read from output connections and passed to the

output device routines, but no I/O is done. The LTS remains in start-up mode until two conditions are fulfilled:

- Each output connection has received start_count bytes of data, as described above, and
- Each output device has been given enough data to begin display (for example, a video decompression system may require several frames "in the pipeline" before display can start).

When these conditions hold, I/O starts on the LTS's logical devices. Output begins on each logical device at the next physical-device interrupt; skew on start-up is therefore bounded by the longest interrupt period. When a new output connection is bound to an already-running LTS, its pipelines are filled in a similar manner prior to starting output at the given LTS offset.

Rate matching. The rate of a running LTS is determined by one of the methods discussed above under "LTS operations and parameters," while the rate of each of its logical devices is determined by the associated physical device. These rates may differ slightly. To maintain synchrony, several approaches are possible:

- (1) vary the hardware I/O rate of the physical device;
- (2) interpolate data to adjust the effective I/O rate;
- (3) skip or pause the logical device (see the sidebar, "Rate adjustment by skipping and pausing").

Approaches (2) and (3) have the advantage that, in the case where a physical device handles streams from different LTSs, the rate of each stream can be adjusted separately. The advantage of (3) over (2) is that it is computationally less expensive.

For LTSs that are device-driven (say, by D_{master}), Acme adjusts rates by skipping and pausing other logical devices in the LTS; D_{master} does not skip or pause. On each interrupt of another device in the LTS (call it D_{slave}), Acme must decide if skipping or pausing is needed. The algorithm used is as follows: Let P_{master} and P_{slave} denote the nominal interrupt periods of D_{master} and D_{slave} respectively, and let T_{master} and T_{slave} denote their actual interrupt periods. Let X denote the rate of D_{slave} relative to D_{master} that is,

$$\left(\frac{P_{\text{slave}}}{T_{\text{slave}}} \right) / \left(\frac{P_{\text{master}}}{T_{\text{master}}} \right)$$

and let $\bar{X}$ denote an a priori upper bound on $\max(X, 1/X)$, as determined

Rate adjustment by skipping and pausing

Acme adjusts the rate of logical devices by *skipping* and *pausing*. Skipping speeds up a device relative to real time, while pausing slows it down. For an output device, the device-interrupt routine skips by discarding data from the output buffer; this can be done only if there are at least two blocks in the buffer, since one block is needed for display in the upcoming period. To pause, the interrupt routine does not remove any data from the buffer, and displays "null" data (repeating the previous video frame, or playing zero audio samples) for one period. For input devices, skipping is done by writing null data to the buffer, while pausing is done by discarding the current block instead of writing it to the buffer.

The timing of I/O on a logical device can be represented using a *stair-step diagram* (see Figure B). Each box represents the I/O of one block of data. The sides of a box are real times at which its I/O starts and ends (that is, the times of consecutive I/O interrupts from the physical device). The bottom and top are the start and end time stamps of the data. The stair-step boxes define a square if the device runs exactly at its nominal rate, as in (a); otherwise they define an oblong rectangle as in (b) and (c).

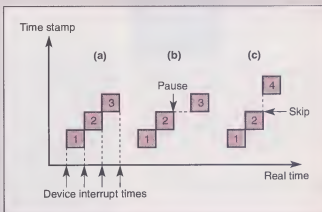


Figure B. Stair-step diagrams for logical devices with pausing (b), skipping (c), and neither (a). The diagrams show the relation between real time and the time stamp of data being input or output on a particular logical device.

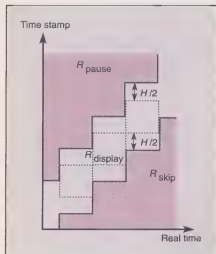


Figure 3. A graphic representation of the skip/pause algorithm used to control the rate of a logical device D_{slave} in a device-driven LTS. The dotted lines are the stair-step diagram of the master logical device D_{master} . The shaded regions represent situations in which D_{slave} is skipped or paused.

by hardware tolerances. Let $H = P_{\text{slave}} \bar{X}$. Extend each block of D_{master} by $H/2$ in both the upward and downward directions, defining regions R_{display} , R_{pause} , and R_{skip} as shown in Figure 3.

On each interrupt of D_{slave} , Acme sees which region (R_{display} , R_{pause} , or R_{skip}) contains the upper right corner of the data block just handled, and acts accordingly. Figure 4 shows examples of the behavior of this algorithm.

If we assume that no buffer overrun or starvation occurs and that physical device rates are constant, the following properties have been proved¹¹ for the algorithm:

(1) The skew between D_{slave} and D_{master} is bounded by

$$\frac{3H}{2} + (\bar{X} - 1) \max(P_{\text{master}}, P_{\text{slave}})$$

(2) Skips and pauses are not interspersed for any logical device. That is, if both skips and pauses occur, one type strictly precedes the other.

(3) If $\bar{X} < 1$ then the fraction of skipped blocks of D_{slave} approaches $(1 - \bar{X})/\bar{X}$. If $\bar{X} > 1$ then the fraction of pause approaches $(\bar{X} - 1)/\bar{X}$.

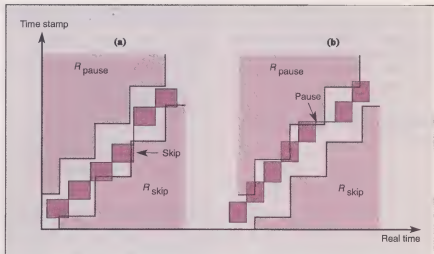


Figure 4. Two examples of the skip/pause algorithm for rate adjustment. The stair-step diagram for D_{slave} is shown using dark shaded boxes. In (a), D_{slave} runs slower than D_{master} so it must eventually skip data; in (b), D_{slave} runs faster than D_{master} so it pauses.

$\bar{X}$ is typically close to one, so the skew bound in (1) is close to

$$\frac{3P_{\text{slave}}}{2}$$

For example, if D_{slave} is a 30 frames per second video device, then its skew will not exceed about 50 milliseconds. Property (2) says that the algorithm avoids needless skipping and pausing, and (3) says that little skipping and pausing is done if devices run at about the same rate.

Buffer starvation/overrun. If a logical device blocks (due to buffer starvation or overrun), it will fall behind its LTS. As long as the skew does not exceed skew_max , I/O proceeds normally on other logical devices in the LTS. The rate-matching algorithm will skip data when possible, and the logical device may catch up to the LTS, as shown in Figure 5a.

If the skew of a logical device D exceeds skew_max , the LTS is temporarily stopped: Its value does not change, and no I/O is done on the logical devices

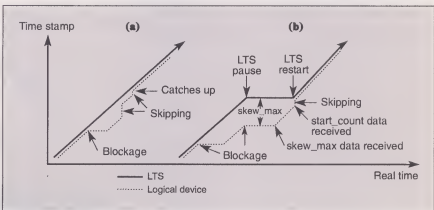


Figure 5. Two blockage scenarios. In (a), the blockage is cleared before skew exceeds the bound skew_max . In this case, data is displayed or skipped, and the logical device catches up to the LTS. In (b), skew_max is exceeded, and the LTS is stopped. It is restarted only when enough data is received for the blocked logical device to catch up, plus start_count additional data bytes.

bound to it (see Figure 5b). If D is an output device, then while the LTS is stopped, data is read from D 's connection until it has received enough to bring D up to the current value of the LTS. At this point, D resumes I/O. Once the connection has received an additional start_count bytes of data (to guard against future blockage), it restarts the LTS and skips if necessary. If D is an input device, null data is written to bring D up to the current value of the LTS, and the LTS is restarted when the connection drains start_count bytes of data from the buffer.

Acme provides a set of abstractions for continuous media I/O. It has been used to implement several applications, including a conferencing system and an A/V recording system. Our experience with Acme suggests two shortcomings in its design:

- (1) CM data within a server can be routed only between network connections and I/O devices; a more flexible architecture might allow connections directly between logical devices (for local echo) or between an output "mixer" and a network connection (for example, for making a recording of a telephone call).
- (2) The LTS abstraction specifies local synchronization only; it cannot easily specify certain requirements on global synchronization (for example, that two users engaged in a telephone conversation should hear a sound file synchronously).

More generally, the Acme programming interface provides low-level abstractions (connections, virtual devices). For writing distributed applications such as a conferencing system, a higher-level model would be preferable (for example, one that hides the connection topology by which audio data is mixed and distributed).

The Acme implementation of logical time systems could be improved or refined in various ways. For example, the rate adjustment algorithm could use information about the time of I/O interrupts to skip or pause at a finer granularity than a whole block, thus improving the bound on skew (the current

algorithm considers only the order of interrupts, not the times at which they occur). ■

Acknowledgments

Pamela Chan, Ramesh Govindan, and Cindy Hertzner contributed to the design and implementation of Acme. Diane Wegner provided the discipline.

References

1. B. Arons et al., "The Voice and Audio Server for Multimedia Workstations," *Proc. Speech Tech 89*, May 1989, pp. 86-89.
2. P. Zellweger, D. Terry, and D. Swinehart, "An Overview of the Etherphone System and Its Applications," *Proc. Second IEEE Conf. Computer Workstations*, IEEE, New York, 1989.
3. L.F. Ludwig and D.F. Dunn, "Laboratory for Emulation and Study of Integrated and Coordinated Media Communication," *Proc. SIGCOMM 87*, ACM, New York, 1987, pp. 283-291.
4. L.F. Ludwig, N. Pinciver, and M. Cohen, "Extending the Notion of a Window System to Audio," *Computer*, Vol. 23, No. 8, Aug. 1990, pp. 66-72.
5. A. Hopper, "Pandora — An Experimental System for Multimedia Applications," *Operating System Review*, Vol. 24, No. 2, Apr. 1990, pp. 19-34.
6. R.W. Scheifler and J. Gettys, "The X Window System," *ACM Trans. Graphics*, Vol. 5, No. 2, Apr. 1986, pp. 79-109.
7. B.D. Markey, "Emerging Hypermedia Standards: Hypermedia Marketplace Prepares for HyTime and MHEG," *Proc. 1991 Summer Usenix Conf.*, pp. 59-74.
8. R. Terek and J. Pasquale, "Experiences with Audio Conferencing Using the X Window System, Unix, and TCP/IP," *Proc. 1991 Summer Usenix Conf.*, pp. 405-417.
9. S. Angebrannndt et al., "Integrating Audio and Telephony in a Distributed Workstation Environment," *Proc. 1991 Summer Usenix Conf.*, 1991, pp. 419-434.
10. G.M. Woodruff and R. Kostpaiboon, "Multimedia Traffic Management Principles for Guaranteed ATM Network Performance," *IEEE J. Selected Areas in Comm.*, Vol. 8, No. 3, Apr. 1990, pp. 437-446.
11. D.P. Anderson and G. Homsy, "Synchronization Policies and Mechanisms in a Continuous Media I/O Server," Tech. Report No. UCB/CSD 91/617, UC Berkeley, 1991.



David P. Anderson is an assistant professor in the Computer Science Division at the University of California at Berkeley. He has done research in distributed operating systems, software support for digital audio/video, computer music, distributed programming, computer graphics, and protocol specification.

Anderson received a BA in mathematics from Wesleyan University. He received an MA in mathematics and an MS and a PhD in computer science, all from the University of Wisconsin-Madison.



George Homsy is a researcher on the Acme project at the University of California at Berkeley. He also works with Survival Research Laboratories, a San Francisco industrial performance art group. He has previously worked as a biomedical instrumentation designer and an independent software consultant.

Homsy received his BS in electrical engineering and computer science from the University of California at Berkeley in 1987.

The authors can be reached at the University of California at Berkeley, Dept. of Electrical Engineering and Computer Science, Berkeley, CA 94720.

InternetExpress: An Inter-Desktop Multimedia Data-Transfer Service

Murugappan Palaniappan and George Fitzmaurice, Brown University*

Internet users today need to share multimedia data like text graphics, spreadsheets, and images as much as they need to share ASCII text.¹ Currently, multimedia documents are often transmitted in printed form via fax and express mail. However, sharing electronically stored data in electronic form is more efficient than in printed form. Accessing and modifying an electronic version is faster and more convenient than sharing and annotating a printed copy.²

Consider the following scenario:

Joe and Mary are co-designing the schematic layout of a new computer chip. They work at different sites and use different but compatible CAD/CAM software packages. Joe works on an IBM PC using the L-Edit package in San Jose, California, while Mary works on a Sun workstation using the Magic package in Washington, D.C. Joe finishes one component of the design and transfers it to Mary to check for consistency with her design. A single command from within the application completes the transfer transaction. At the time of Joe's transmission, Mary has already left her office for the day. When she arrives at work the next morning, she sees that she has received an electronic express package from Joe. When she opens it, Joe's component is presented to her in Magic format. She finds something in Joe's design that does not match their proposed specifications. She modifies the design accordingly and telephones Joe to discuss her proposed changes. As their conversation starts, she selects the modified component and sends it to Joe. In a few seconds, a message on his monitor indicates that Mary's package has arrived. He views it immediately in L-Edit format, agrees with the changes, updates his copy of the document, and resends it to Mary.

This scenario illustrates many of the features needed in a data-transfer service. Users who work with compatible applications, possibly across heterogeneous platforms, should be able to share data easily and quickly. Regardless of the application, users should be able to send data by issuing a single command. They often need to send data at the selection level rather than at the document level. The responsibility for determining a compatible application for presenting the data should fall on the service and not on the user.

To date, sharing ASCII text has been achieved primarily by electronic mail. This

Featuring a simple desktop operation that works uniformly across all applications, the InternetExpress integrated service enables users to send and receive selection-based multimedia data in a timely manner.

* Since collaborating on this article, Palaniappan joined Aldus Corp. and Fitzmaurice started doctoral studies at the University of Toronto.

tool is quite independent of any application and often requires the author to directly enter the ASCII text using a simple editor before sending. Indeed, the success of electronic mail can be attributed to its widespread availability across heterogeneous environments, its simple interface, its reliability, and its low cost.

Sharing multimedia data, however, is inherently more difficult than sharing ASCII text. Unlike ASCII text, multimedia data is created within an application (for example, pictures are most likely created within a graphics application). In current practice, these applications have not been designed to transport the data to users at different sites. Instead, network-based tools, such as file-transfer protocol (FTP), and phone-based tools, such as facsimile machines, have been used for transferring the data.

Most existing tools fall short in one or more of the following areas:

- *heterogeneity* — the ability to work across different hardware and software platforms;
- *seamlessness* — the ability to work uniformly across all applications;
- *user-interface simplicity* — ease of use, requiring few steps for a single transfer;
- *timeliness* — the ability to transfer data almost instantaneously; and
- *security* — preventing outside users from accessing restricted data.

To overcome the shortcomings in existing network-based transfer tools, we have designed an integrated desktop data-transfer service called InternetExpress. We chose the name InternetExpress to denote quick, reliable, and timely transfer of multimedia data on the Internet. This service allows users to share data conforming to any data standard across heterogeneous environments on the Internet, assuming the environments follow the general principles of the desktop metaphor.^{3,4} InternetExpress is selection-based and works seamlessly across all desktop applications. A prototype of InternetExpress has been implemented in the Intermediate desktop environment.⁵

Background

Before describing the InternetExpress service, we critique numerous solutions

InternetExpress lets users share data conforming to any data standard across heterogeneous environments on the Internet.

currently used for transferring multimedia data. Then, we summarize the characteristics that make these solutions successful.

Related network-based solutions.

Many network-based tools such as FTP and Kermit have been developed to transfer data across wide area networks. But we believe these tools fall short in that they impose burdens on the users and raise security concerns.

A user usually cannot deposit documents in another user's directory at a remote site unless the user has access rights to connect. Consequently, a sender initiates the transfer by asking a recipient to retrieve the document at a specified location. The sender must also set up permissions so that the recipient can connect to the sender's system to retrieve the document.

To set up permissions, the sender could disclose his or her user password or get the system administrator to set up a special, restrictive account for the recipient. Once connected, a recipient is usually able to browse other documents on the system and has the potential ability to modify or deposit documents.

After retrieving the document, the recipient still might not be able to interpret the data easily. For instance, a sender could save an Excel spreadsheet document in an industry data standard such as WK1 and inform the recipient of this. If the recipient does not have an Excel application, he or she has to know that it is still possible to view the data using a Lotus 1-2-3 application that reads the WK1 file format.

A second approach to sharing documents is to rely on a remote file system mounting capability, as in the Andrew File System.⁶ Like the previous approach, this solution places many burdens on users and raises security issues. By sending information about the loca-

tion of a document (via electronic mail, for example), a sender can ask a recipient to remotely mount the file system on which the document resides and either access or make a copy of the document.

For a file system to be remotely mounted, the sender must get the system administrator to grant permissions to the relevant sites. The system administrator has to update one or more system files, both when granting and when removing access privileges. At the recipient's site, the system administrator must mount and unmount the file system, as necessary. Once the file system is mounted, a recipient can access any documents on the file system as though they are on a local file system. Again, a recipient may not be able to easily interpret the accessed or retrieved data.

In both approaches, the sender is more likely to initiate the transfer transaction. This can present a problem if a receiver does not retrieve the data immediately. For example, the document could be moved, renamed, or modified.

In another approach, a user on a Unix-based machine can send multimedia data as an electronic mail message. This solution requires the sender to convert any binary multimedia data into ASCII form using the *uuencode* command. Once in ASCII form, the data can be included as part of the message. The recipient then has to extract the data from the message and/or restore it as binary multimedia data using the *uudecode* command. This approach is cumbersome and involves multiple steps.

A fourth approach developed recently by several computer vendors lets users transfer multimedia documents conforming to the data formats within their specific packages. For example, BBN/Slate, IBM/Office Vision, HP/NewWave Office, and DEC/All-in-One are similar packages that contain a suite of applications for creating and sharing multimedia documents. In most of the packages, the applications display only the data, while relying on electronic mail to physically transfer the data.

These self-contained packages let users transfer multimedia data only to sites that have the specific package installed; each package imposes restrictions on adding new data types. A recipient who does not have the package can obtain an ASCII text version of the multimedia documents. For example, an image would be represented as one

line of text stating that the picture would have appeared at that location. The primary limitation of this approach is that the transfer mechanism is part of a self-contained package and is not fully integrated into each vendor's desktop.

A fifth approach is to use vendor-specific, enhanced electronic mail systems, such as Microsoft/MS Mail and CE Software/QuickMail. Both systems follow the Macintosh-style graphical user interface and are easy to use. While sending an electronic mail message, QuickMail users can include a selection previously placed on the clipboard. When the recipient reads the message, the accompanying multimedia data is placed on the recipient's clipboard. Using a clipboard, however, is problematic, since it prevents a user from sending multimedia data to people who do not have clipboards, and it destroys the resident contents of the recipient's clipboard.

Both systems are designed to operate on local area networks and need modems to communicate with other sites, thereby restricting data-transfer timeliness. In addition, these services require the recipient to know the appropriate application to use to interpret the incoming data.

Although it is only an ASCII data-transfer mechanism, the Unix *talk* program is a sixth approach that is architecturally interesting. This utility allows users to communicate across heterogeneous machines on the Internet by sharing data in real time. A user can initiate the communication in a single step and be notified of the presence or absence of other parties.

This program uses point-to-point connections as opposed to the store-and-forward approach used by electronic mail. Users, however, can only communicate with others who are active at their workstations or terminals. Further, the utility uses an abrasive notification mechanism to inform other parties of the communication request.

Related technologies. Facsimile machines' transfer bitmap images of multimedia data over telephone lines. The image transmission takes place over a high-bandwidth, point-to-point connection. These dedicated fax machines have been successful because they offer fast communication, use the existing telephone-line infrastructure, have a simple user interface, and can send any

handwritten or printed material. But, since a fax is sent to a machine, usually a shared resource, there is often a delay between arrival at a machine and receipt by the individual.

One improvement to using a fax machine is to add a fax modem to a computer. The fax modem automates document transmission directly from the computer, eliminating the need to print a paper copy. Examples of fax modems include Apple's AppleFax, Abaton's InterFax, and STF Technologies' Fax-STF. These modems let you store faxes in a proprietary format to be sent to a fax machine or a computer equipped with a fax modem, but they do not support industry data standards or provide imaging or manipulating capabilities.⁴ Extending the AppleFax software with Solution's BackFax software allows background transfer of not only faxes but also binary and text files. The files, however, can only be transferred to sites that have an AppleFax modem and BackFax software.

Vendor-specific networks such as AppleLink can be considered another approach to document transfer. AppleLink consists of a modem-accessed telephone network and an application to create, send, and receive documents. The AppleLink network is connected to a mainframe that deals with document routing and storage. Registered users can send ASCII text messages or Macintosh files to AppleLink users. This approach works well among Apple computers but can't work across heterogeneous systems.

Solutions International's Super Glue II provides a partial solution to the problem of sharing multimedia data by defining a standard Glue format and supplying a system-viewer application. The utility captures the "printed output" of a document on disk (in Glue format) instead. This Glue file can then be sent using standard transport mechanisms such as AppleLink. The recipient of a Glue file can view the data with the system-viewer application. This utility does not preserve the original format of the data, thereby preventing a recipient from manipulating the data. Converting the data to Glue format is a multi-step procedure.

Although it does not directly address the problem of sharing multimedia data, Software Innovation markets an interesting Macintosh file-launching utility called HandOff II, which allows users

to work with foreign file formats. In the Macintosh environment, double-clicking on a data file brings up the application in which the data was created. If the application does not exist, an error message is presented; the user must launch a compatible application that can interpret the data.

The HandOff II utility provides a mechanism for users to bypass this procedure. For example, a user may not be able to open a file created by a Lotus 1-2-3 application. The utility, however, would recognize the WKS format and present the data in a compatible application like Excel.

Another popular method of sharing multimedia data is to use overnight delivery services such as Federal Express. Since this method is based on postal addresses, documents can be transferred reliably to almost anyone. This service burdens the user, who has to print and package the document, fill out the necessary forms, and abide by rigid drop-off times for sending and receiving documents.

Important characteristics of data-transfer mechanisms. After surveying the data-transfer mechanisms, we recognized characteristics that made each approach successful. We then identified 12 important characteristics that we believe constitute a complete solution to transferring multimedia data. Table 1 summarizes how each system holds up to these characteristics. Each of the surveyed systems falls short in two or more categories. Most notably, the data-transfer mechanisms often have cumbersome interfaces, lack software and hardware heterogeneity, and separate the transport mechanism from the applications.

We also recognize that cost is an important factor in determining the frequency of use of a data-transfer system. Comparing the cost of transferring data across different technologies, however, is beyond the scope of this article. In general, network-based technology is relatively cheaper than phone-based tools and overnight delivery services. Users can afford to transfer data frequently over the Internet domain because it is currently free for most users; the government covers the cost involved in both transferring the data and maintaining the network.

We developed InternetExpress by considering the strengths of some of the

Table 1. Characteristics of data-transfer mechanisms.

Name of System	Fine Granularity	Seamlessness	Hardware Heterogeneity	Software Heterogeneity	Timeliness	Single Step	Preserve Format	Notification	Confirmation	Enclosed Message	Security	Wide area Network
FTP/Kermit			Y				Y					Y
Andrew File System		NA	Y	NA	Y	Y(1)	Y					Y
Unix E-mail		NA	Y	NA			Y	Y		Y	Y	Y
BBN Slate			Y				Y(6)	Y		Y	Y	Y
Office Vision			Y(2)				Y	Y	Y	Y(7)	Y	Y
NewWave Office							Y	Y(8)	Y(8)	Y	Y	Y
DEC All-in-one			Y				Y	Y	Y	Y	Y	Y
QuickMail	Y	Y(3)			Y(4)		Y	Y	Y	Y	Y	Y(9)
Microsoft Mail		Y(14)	Y		Y(4)		Y	Y	Y	Y	Y	Y(9)
Unix <i>talk</i>	NA	NA	NA	NA	Y	Y	NA	Y	Y	NA	Y	Y
Fax		Y(5)	NA	NA	Y					Y	Y(10)	Y
Glue with Apple Link	Y	Y	Y	Y				Y			Y	Y(9)
Apple Link							Y	Y	Y		Y	Y(11)
Federal Express			NA	NA			Y(12)	Y	Y	Y	NA	Y(13)
Internet Express	Y	Y(14)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Definitions												
Fine granularity:			Can data at the selection level be transferred?									
Seamlessness:			Can all desktop applications participate in the systemwide service?									
Hardware heterogeneity:			Is the data transferrable across different hardware platforms?									
Software heterogeneity:			Can the desktop automatically launch applications based on data type?									
Timeliness:			Is the data transferred directly from sender to receiver in a single step?									
Single step:			Does it take only one command for the user to send or receive data?									
Preserve format:			Does the transfer maintain the original electronic format of the data?									
Notification:			Is the user informed when data arrives?									
Confirmation:			Can a sender ask to be informed when a recipient receives the data?									
Enclosed message:			Can a message be enclosed along with the data being transferred?									
Security:			Is the receiver prevented from browsing/modifying documents in the sender's file system?									
Wide area network:			Can data be transferred to any site on the Internet?									
Notes												
(1) Initially, requires many steps to connect. (2) Works only across IBM PC compatibles. (3) Uses a system clipboard for the transfer. (4) Achieved by using a modem. May be a problem if modem is already in use. (5) Fax boards allow applications to participate uniformly by using the print mechanism. (6) Writes and transfers only vendor-defined multimedia data. (7) Fixed-length message. (8) Depends on underlying electronic mail service. (9) Needs a modem to connect to other local area AppleTalk networks. (10) An issue at the receiver's end, since one user may be able to pick up another user's fax. (11) If AppleLink has access to Internet. (12) Only physical entities can be preserved. (13) Uses postal addressing scheme. (14) If the applications comply with the APIs.												

more prominent data-transfer mechanisms. We wanted a mechanism akin to Unix electronic mail that would transfer multimedia data over heterogeneous systems across the Internet. Unlike electronic mail's store-and-forward technique, we opted for a direct, single-step

point-to-point transfer method, analogous to fax machines and Unix *talk*, to ensure fast and reliable delivery. Our approach, unlike Unix *talk*, does not require recipients to be available or logged on for the transfer to occur. We borrowed the cover-sheet concept from

fax machines to allow users to enclose text messages and specify one or more recipients.

Like MS Mail, we defined an application programmer's interface to extract data from applications. We further extended the API to present incoming

data in an appropriate application, a process similar to the functionality of HandOff II. Also, we believe users want to share data at the selection level, as in QuickMail.

Following the conventions of several data-transfer mechanisms, we provide notification and confirmation in our solution. InternetExpress borrows the notion of packages used by overnight delivery services and provides notification by using an express-package icon that sits on the user's desktop. Like OfficeVision and All-in-One, our system lets users track the status of a sent message.

Overview

With our architecture, selections of any type of data can be sent across the

Internet and received within seconds by any machine that implements the InternetExpress service. Since our service is based on sending, not retrieving, security is not an issue. Because InternetExpress operates on the Internet, data transfer is as inexpensive as electronic mail.

We have implemented an InternetExpress prototype in the Intermedia desktop environment and tested it across different local area networks at Brown University. Intermedia applications that support this data-transfer service include InterDraw, a structured graphics editor; InterWord, a word processor; and InterVal, a timeline editor. InterDraw and InterVal write their data in PICT format, while InterWord writes its data in RTF format.

InternetExpress allows data sharing at both the selection and document lev-

els. For example, a user may want colleagues to review only certain sections of his document. With only document-level granularity of data transfer, the user must select the relevant sections and copy the data to a buffer, and then paste the data into a new document for transmission. With selection-based transfer, the user need only make a selection and send the data.

Within the Intermedia desktop environment, the data-transfer functionality is integrated into each application, so the actions of selecting and sending the corresponding data work uniformly across all applications.

To achieve this level of integration, the service defines three abstract methods (sendData, receiveData, and understandData) that an application must implement to participate in the service. The service, not each application, is responsible for sending, routing, and storing the data. This paradigm is modeled after the ubiquitous cut/copy/paste paradigm and can be viewed as placing data on and retrieving data from a network-wide, remote clipboard or buffer.

Because many people are familiar with express-mailing packages, our data-transfer service uses an express-package metaphor. The metaphor of expressing a package suggests that users can send and receive multimedia data quickly, reliably, and easily from their desktops.

Our service allows users to share data across heterogeneous environments. We believe this characteristic is crucial, as does Kraut.⁹ In a CHI 90 panel address, he argued that any groupware tool that holds allegiance to a single hardware or software base is likely to result in failure. As a consequence, InternetExpress is composed of a desktop-independent component and a desktop-dependent component. The desktop-independent component is a background process, or daemon, that continuously checks and receives incoming data from users on any hardware platform. A user can then receive the data from any desktop that supports our service.

Since each desktop may not have access to the same set of applications, the desktop-dependent component is responsible for choosing a compatible application to present the data to the user. For example, a package of a SuperPaint document from one site could be sent to a remote site that only has MacDraw, since both MacDraw and Super-

Desktop requirements

Before describing a set of desktop requirements to support a data-transfer service, we must agree on a minimal desktop definition.

First, it is necessary to distinguish between functionality provided by an operating system and that provided by a desktop. The desktop provides an extra layer between the user and the operating system. The user perceives this desktop layer in the form of a graphical user interface for accessing and modifying system resources (for example, files, printers).

As the user interacts with the desktop resources, the desktop can monitor and track the activity to provide support above and beyond that of the operating system. For example, the desktop could monitor a user's window layout and restore it the next time the user logs into the desktop.

In terms of functionality, a minimal desktop should have a graphical file manager that allows the user to browse, access, and manage files. More sophisticated desktops will be able to associate applications to data files so that when the user opens a document the proper application is launched to receive and display the document. The desktop should have a set of system commands that are available to the user at all times, usually in the form of menus. Some common commands are printing, opening

and closing files, and bringing up system tools.

Environments such as Apple's Macintosh, Sun's OpenWindows, and Microsoft's Windows are examples of desktops that satisfy the minimal definition. A vanilla Unix or DOS environment does not by itself constitute a desktop because it lacks the extra layer of functionality between the user and the operating system resources.

In addition to the minimal definition to support InternetExpress, the environment must be enhanced to handle user identification and user notification. User identification is important, since it is often more convenient to address and receive information on a user basis rather than on a machine basis. The degree of user notification should be commensurate with the importance of the information. For example, extremely important information could warrant interrupting the user by displaying a dialogue box, while less important information could be presented to the user by a visual cue, such as a modified icon. In any case, a user should be able to customize the notification mechanisms used.

In summary, a desktop that can implement our service should have user identification and notification, in addition to the minimal desktop definition described above.

Paint support the PICT data type.

To facilitate sharing multimedia data among users across heterogeneous environments, Straub and Wetherbe¹⁰ and Borenstein¹¹ argue that applications should conform to industry data standards (for example, PICT, RTF, and WK1). We agree and argue further that a desktop service should not restrict new data standards. If a new data standard is formalized, the service should not have to be upgraded.

InternetExpress supports the transfer of data conforming to any data standard because it is responsible for packaging and unpackaging the information accompanying the data (that is, application type, data type) that describes the selection.

In the "Desktop requirements" sidebar, we enumerate additional desktop functionality needed to implement our service.

User interface

Next, we describe how express packages are sent and received using the InternetExpress interface.

Sending an express package. To send an InternetExpress package, a user makes a selection and then fills out an express cover sheet to accompany the data. If a selection is made within an application, the contents of the selection will be sent. Alternately, if a user selects a document icon, the entire contents of the document will be sent. If there is no selection, then only the express cover sheet is sent.

Once a selection is made, the user picks the Send Express Package desktop menu command. This action displays an express cover sheet, as shown in Figure 1.

Various options let the sender quickly classify the status of an express package. The first setting allows the sender to include or exclude the selected data or document. To request the recipient's immediate attention, the sender needs to mark the express package as urgent. The third option is a quick way for the sender to indicate whether or not a reply is expected.

Figure 1. Sample InternetExpress cover sheet.

To request confirmation when a recipient has seen the package, the sender selects the Confirm Receipt setting. This setting automatically generates and transmits a confirmation notice when a recipient opens, reads, or deletes the package.

Receiving an express package. We have designed — but not yet implemented — an iconic interface for receiving express packages. The icon reflects the state of the spooled express packages, as shown in Figure 2.

A user can receive an express package at anytime. Packages that arrive while a user is not logged on are spooled. When a user starts an Intermedia session, an audible beep and the icon's visual state announce the presence of new packages.

If an express package arrives during an active session, the user will be notified in a variety of ways, depending on the state of the spooled packages. If regular or confirmation packages arrive, the current package icon flashes and changes to reflect the state of the new arrival (see Figures 2c and 2d). If audio notification is turned on, a double beep indicates the arrival of a regular package, while a single beep indicates the arrival of a confirmation package.

For a regular package marked urgent, the package icon (see Figure 2d) flashes periodically in case the user missed the beep. This periodic notification stops when the user responds by viewing the package or by clicking once on the icon. The user has the option of turning off both the beep and periodic notification.

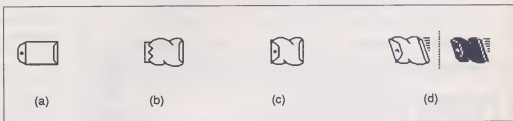


Figure 2. The package icon appears on the user's desktop and indicates the state of the spooled express packages: (a) no packages for the user, (b) one or more read packages are still spooled, (c) one or more new packages exist, (d) at least one new, urgent package exists.

To receive a package, a user selects the Read Express Package menu option or double-clicks on the package icon, which brings up an Express Arrivals bulletin board that displays a list of spooled packages, as shown in Figure 3. The list entries are sorted by time, with the most recently received package at the top.

Each entry consists of a one-character status indicator, followed by the sender's name. A trailing number helps distinguish multiple packages from the same sender. The indicators are "U" for an urgent package, "P" for a regular package, and "C" for a confirmation notice package. If a user has read a package, the indicator appears in gray.

To operate on a regular express package, a user selects an item from the list and then chooses one of the three buttons: View Data, Delete, or Get Information. The View Data button will retrieve the package data and display it in an application that understands the data type. To delete a package, the user selects the desired item in the list and chooses the Delete button. By choosing the Get Information button, the user can examine the express package cover sheet, as shown in Figure 4.

As a short-cut, the user can double-click on a package entry in the bulletin board list. With this action, the express cover sheet appears, and the package data appears as a new document. Express package documents function like

other documents on the desktop. For example, a user can rename and save the package documents into any folder. The cover sheets are discarded when the documents are saved.

Confirmation notices, which look similar to cover sheets, can be opened by selecting a confirmation notice from the bulletin board and choosing the View Data or Get Information buttons or by double-clicking on the entry. This notice indicates the recipient's name, the date the package was sent and read, and the name of the source document.

Architecture

To support the interface described above, InternetExpress uses a TCP/IP (transport connection protocol/Internet protocol) point-to-point connection to send express packages to sites on the Internet. The service consists of a desktop-independent component and a desktop-dependent component.

The desktop-independent component

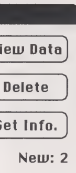


Figure 3. The InternetExpress arrival bulletin board.

is an express daemon that services a given local area network. The daemon runs at an advertised port on a server machine (analogous to an electronic mail daemon) and is responsible for receiving packages on a user's behalf. Since our service uses the same Internet routing tables as electronic mail, the daemon must run on the same machine as the

mail daemon. In this way, InternetExpress is as adaptable to Internet changes as electronic mail.

The desktop-dependent component, the client, defines a protocol for participating applications. It is also responsible for packaging and unpackaging data and for mapping the data to an application. We have implemented the Intermedia desktop-dependent component on the Macintosh family of computers running A/UX, and the daemon on a Sun workstation running SunOS 4.03.

Application programmer's interface.

Our Intermedia design philosophy has been to provide general services at the desktop level and to define abstract methods that applications must implement to participate. For instance, linking services in Intermedia define generic methods for applications to participate in the hypermedia functionality.^{8,12}

Following our design philosophy, we defined three methods for Intermedia application participation in our desktop data-transfer service:

- DoWriteData,
- DoReadData, and
- DoUnderstandData.

Once a user is ready to transmit, the client invokes the application's DoWriteData method, which returns a data type (for example, PICT, RTF) and data replicating the selection.

When a user is ready to view a package, the client chooses an application and calls its DoReadData method to display the selection.

During application installation into the desktop, the client invokes the application's DoUnderstandData method, which returns all the data types that it can understand.

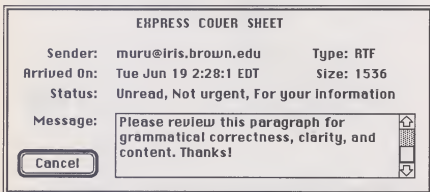


Figure 4. Another sample InternetExpress cover sheet.

Other desktops that wish to implement the express service will need to implement similar methods.

Express daemon. Express packages are not sent directly to a user but rather to an express daemon, which then spools them in the local file system. An express daemon is initialized with a spool directory on the file system that will store incoming packages (equivalent to /usr/spool/mail). Every user has a subdirectory in the spool directory for incoming packages. Each package is saved as a separate file.

An express package file consists of two sections: a header and the application data. The header section contains an application-type field, a data-type field, and a collection of fields representing the cover-sheet information, including a status field (that is, read/unread, awaiting reply/FYI, etc.). The client determines the application type from which a selection has been sent.

The data-type field is set based on the information returned from calling an application's DoWriteData method. The application data corresponds to the actual data needed to reconstruct a selection. As all packages have standard header information, any desktop can examine the package and determine the application for presenting data to the user. The desktop can inspect the package's status field to ascertain its state.

An express daemon accepts connections from any host and provides local area network services that include Internet domain name resolution and package delivery to local clients. To understand the domain name resolution service, consider two local area networks connected by the Internet, as shown in Figure 5.

If Joe wants to send an express package to Ann, his sending process sends it to the local-domain express daemon, which then spools the data for Ann in the file system. When Joe wants to send an express package to Tom, the sending process first obtains the Internet address of the foreign-domain daemon from the local daemon and sends the package directly; the package is then spooled for Tom on his local file system. It is important that the daemon provide this service, since it obviates the necessity for each host to maintain an up-to-date listing of all daemons on the Internet.

As information arrives, the desktop-

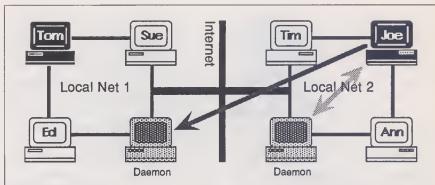


Figure 5. Joe wishes to send an express package to Tom in a foreign domain. The sending process gets the foreign daemon's Internet address from the local daemon and sends the package.

dependent component or client must be notified. This is achieved by having the express daemon server send a message to the clients that match the recipient's name on the express package. When a user logs into a desktop, the client sends a message informing the express daemon of the user's location. The client-daemon model eliminates the need for client machines to mount a shared express directory and poll for the arrival of new packages. This scheme allows sending packages to clients on heterogeneous machines, since the data is sent using a common communication protocol, TCP/IP.

Desktop client database. Each client maintains a database of all the applications that can run on the desktop, together with the data types they under-

stand. Clients who share a file system maintain a shared database. The database information is maintained in two relational tables. One lists data types with applications that understand each data type, as shown in Table 2.

The second table lists applications and their locations in the file system, as shown in Table 3. Several desktops, such as the Macintosh and OpenWindows, have a "finder" mechanism to locate an application in the file system. If such a mechanism exists, the desktop does not need to maintain a second table.

A user usually installs an application into the desktop by dragging an application icon and dropping it into a folder. If the user has changed the location of an existing application, the corresponding entry in the second table is modified. On the other hand, if a new application

Table 2. Data type and applications mapping table.

Data Type	Applications
PICT	MacDrawII, SuperPaint 2.0a
WK1	MS Excel, Lotus 1-2-3
SYLK	MS Excel, Wing Z, Lotus 1-2-3

Table 3. Applications and location mapping table.

Applications	Location in File System
MacDrawII	Applications/DrawProgs
SuperPaint 2.0a	User/muru
MS Excel	Applications
Wing Z	Applications

has been installed, the client adds an entry in the second table and tags it as an application whose data types need to be retrieved and added into the first table.

When a user runs a tagged application for the first time or explicitly selects the Install Application menu command, the desktop launches the application and invokes its DoUnderstandData method. The application is responsible for returning a set of data types that it can interpret. For example, a MacDraw II application returns Drawing, PICT, Color PICT2, and Stationery as its data types. The desktop updates the corresponding entries in the first table.

Mapping data types to applications.

When a user is ready to view a package, the desktop client is responsible for receiving the package from the daemon and launching an application that can interpret the data. The client first examines the application type and checks its database to see if the application is avail-

able. If the application is available, it launches it; otherwise, the client examines the package's data type and launches the first compatible application that appears in the database.

If no compatible application exists, two possible options are to allow the user to see an ASCII text version or to install a compatible application. We chose not to implement the first option, since noncontextual media cannot be readily perceived in ASCII form. Instead, we allow the user to view the cover sheet to recognize the data type and have the option of installing an application that can display the package.

Once an application has been found, the client invokes the application's DoReadData method. This method receives the data and reconstructs the selection.

User evaluation

Although our prototype has had limited use, we have gathered some initial

feedback on the system. Users reacted positively regarding the system's ease of use and simple interface. Defining a selection within an application and being able to send it to a recipient was commonly thought of as an extension to the cut/copy/paste metaphor (that is, cut/copy/paste/send).

In terms of functional extensions to the system, a few users wanted to be able to send folders or even entire disks. This is a bit tricky, since there is not really a folder-viewing application to invoke. However, the desktop could take on some of the responsibility to display the folder.

Viewing a package would require the desktop to construct a temporary folder. If a user decided to save the new folder, the desktop would ask the user where to place it relative to the existing folder hierarchy. The InternetExpress service could easily extend the header definition of packages to support folder packages.

A recursive package header definition would allow folders within folders



ANNOUNCEMENT

COCs '91*

CONFERENCE ON ORGANIZATIONAL COMPUTING SYSTEMS

Sponsored by ACM SIGOIS and IEEE CS TC-OA in cooperation with IFIP WG 8.4

Atlanta Marriott Marquis

November 5-8, 1991

**Keynote Address - Bill Curtis, Director, Software Process Program,
Software Engineering Institute, Carnegie Mellon University**

Invited Speakers -

Marciel Losada, Director, Center for Machine Intelligence, Electronic Data Systems Corporation
Jim Foley, Georgia Institute of Technology

General Chair - Fred Lochovsky, HKUST, Hong Kong

SESSIONS:

Organizational Support Systems

Chair: *Murray Mazer, DEC CRL*

Groups Within Organizations

Chair: *Sudhir R. Ahuja, AT&T Bell Laboratories*

Organizational Interfaces

Chair: *Bob Allen, Bellcore*

Objects

Chair: *Peter de Jong, IBM*

Collaboration

Chair: *J. Robert Ensor, AT&T Bell Laboratories*

Coordination

Chair: *Carson Woo, U. British Columbia, Canada*

For information contact Program Chairmen: Sid Ahuja - (908) 949-5569

fax - (908) 949-0399

Peter de Jong - (617) 576-9214

fax - (617) 576-9258

PANELS:

Active Badges

Organizer: *Ken Pier, Xerox PARC*

Wireless Communications

Organizer: *Rich Wolff, Bellcore*

POSTER SESSION

TUTORIALS:

*Dr. Brian R. Gaines, Knowledge Science Institute,
University of Calgary, Canada*

Dr. Charles Grantham, University of San Francisco

*Professor Marilyn M. Mantei, Department of Computer
Science, University of Toronto, Canada*

* Formerly Conference on Office Information Systems

to be sent as one package. Similarly, if a user wanted to send a set of documents having different data types, a virtual folder would be constructed and sent to the recipient.

Some users wanted to be able to send a user face or company logo (in the form of a bitmap) as part of the cover sheet information. We could easily do this in InternetExpress by sending the bitmap data as part of the header information.

Although Intermedia has a small set of applications, users recognized that they would want to specify preferred viewing applications. That is, if more than one application can handle a particular data type, the resolution algorithm should check the user's set of preferences for selecting the viewing application.

In InternetExpress, we have designed a desktop data-transfer service and prototyped it within the Intermedia desktop environment. Based on our design and implementation, we recommend the following characteristics for a similar service in other desktop environments:

- Provide a simple application programmer's interface to allow desktop applications to easily participate in the service.
- Allow applications to register the data types they understand with the desktop so that the transfer service can automatically choose an appropriate application for presenting the incoming data to the user.
- Provide a daemon to receive incoming information on behalf of users so that they can view the data at their convenience from any workstation on the network.
- Support transfer of not only selections within a document but also any desktop selection.
- Have a wide range of notification mechanisms available on the desktop to inform the user of various events.

We believe that InternetExpress serves a real user need, provides a simple user interface, and transfers data quickly and at a low cost to the user. We also suspect that, with widespread adoption of the InternetExpress service, users would share multimedia data as frequently and conveniently as they do electronic mail. ■

Acknowledgments

We especially thank Nicole Yankelovich for her help in refining the concepts described in this article and for her valuable comments on early drafts. We also thank Norman Meyrowitz and Karen Catlin for their valuable critique of the manuscript, and we acknowledge Helen DeAndrade for her assistance in preparing the diagrams.

References

1. T. Crowley, "How Can We Make Groupware Practical?" panel, *Proc. 90 Conf. Computer-Human Interaction*, ACM, New York, pp. 87-89.
2. G. Salton, *Automatic Text Processing*, Addison-Wesley, Redding, Mass., 1989, pp. 471-473.
3. J. Johnson et al., "The Xerox Star: A Retrospective," *Computer*, Vol. 22, No. 9, Sept. 1989, pp. 11-29.
4. B. Shneiderman, "Direct Manipulation: A Step Beyond Programming Languages," *Computer*, Vol. 16, No. 8, Aug. 1983, pp. 57-68.
5. N. Yankelovich et al., "Intermedia: The Concept and the Construction of a Seamless Information Environment," *Computer*, Vol. 21, No. 1, Jan. 1988, pp. 81-96.
6. M. Satyanarayanan et al., "The ITC Distributed File System: Principles and Design," *Proc. 10th Symp. Operating Systems Principles*, ACM, New York, 1985, pp. 35-50.
7. K. McConnell, D. Bodson, and R. Schaphorst, *Fax: Digital Facsimile Technology and Applications*, Artech House Books, Norwood, Mass., 1989.
8. D. Crabb, "Mac Goes Fax," *Byte*, Vol. 14, No. 5, May 1989, pp. 208C-208H.
9. B. Kraut, "How Can We Make Groupware Practical?," panel, *Proc. 90 Conf. Computer-Human Interaction*, ACM, New York, pp. 87-89.
10. D.W. Straub and J.C. Wetherbe, "Information Technologies for the 1990s: An Organizational Impact Perspective," *Comm. ACM*, Vol. 32, No. 11, Nov. 1989, pp. 1,328-1,339.
11. N.S. Borenstein, "Multimedia Electronic Mail: Will the Dream Become a Reality?" *Comm. ACM*, Vol. 34, No. 4, Apr. 1991, pp. 117-119.
12. M. Palaniappan, N. Yankelovich, and M. Sawtelle, "Linking Active Anchors: A Stage in the Evolution of Hypermedia," *Hypermedia*, Vol. 2, No. 1, Summer 1990, pp. 47-66.



Murugappan Palaniappan is a senior software engineer with Aldus Corp. in Seattle, Washington. Previously, he was a research software engineer at Brown University's Institute for Research in Information and Scholarship and the principal architect of the institute's intelligent agent research program. He played major roles in the design and development of the Intermedia system. His major research interests are in the areas of intelligent agents, component-based architecture multimedia systems, and computer-supported cooperative work.

Palaniappan holds an MS degree in computer science from the University of North Carolina at Chapel Hill, an MS degree from Duke University, and a BSc degree in civil engineering from IIT Madras. He is a member of the ACM, SIGOIS, and the IEEE Computer Society.



George Fitzmaurice is a PhD candidate in computer science, working in the Dynamic Graphics Project at the University of Toronto. Previously, he was a research software engineer at Brown University's Institute for Research in Information and Scholarship and a senior researcher and architect of the agent project. He and made substantial contributions to the Intermedia system. His major research interests are in the areas of computer-supported cooperative work, human-computer interaction, and multimedia communication.

Fitzmaurice received his ScM in computer science at Brown University and his ScB in mathematics and computer science at MIT. He is a member of the ACM, SIGCHI, and the IEEE Computer Society.

Readers can contact Palaniappan at Aldus Corp., Engineering Dept., 411 First Ave. South, Seattle, WA 98104, Internet mu@aldus.celestial.com; and Fitzmaurice at the University of Toronto, Dept. of Computer Science, Toronto, Ont., Canada M5S 1A4, Internet g@dgpt.toronto.edu.

IEEE INTERNATIONAL CONFERENCE on WAFER SCALE INTEGRATION-1992

January 22-24, 1992

Fairmont Hotel
atop Nob Hill
San Francisco, California 94106, U.S.A.

Sponsored by the IEEE Computer Society and the
IEEE Components, Hybrids, and Manufacturing Technology Society

General Chair

Vijay K. Jain University of South Florida

European Chair

R. Mike Lea Brunel University

Finance Chair

Joe E. Brewer Westinghouse Corporation

Local Arrangement Chair

Earl E. Swartzlander University of Texas Austin

Program Chair

Peter W. Wyatt MIT Lincoln Laboratory

Asian Chair

Tadashi Sasaki Sharp Corporation

Publicity & Registration Chair

Michael W. Yung Hughes Research Laboratories

Program Committee

Joe E. Brewer, Westinghouse Corporation
W. Kent Fuchs, University of Illinois
Susumu Horiguchi, Tohoku University
Vijay K. Jain, University of South Florida
R. Mike Lea, Brunel University
Michael J. Little, Hughes Research Laboratories
John F. McDonald, Rensselaer Polytechnic Institute
Michael Michael, Westinghouse Corporation
H. G. Musmann, University of Hannover

Ulrich Ramacher, Siemens AG
Mariagiiovanna Sami, Politecnico di Milano
Zoltan Stroll, TRW Corporation
Earl Swartzlander, University of Texas Austin
Stewart K Tewksbury, University of West Virginia
John Toole, DARPA
Jacques Trilhe, SGS Thomson Microelectronics
Peter W. Wyatt, Massachusetts Institute of Technology
Michael W. Yung, Hughes Research Laboratories

The IEEE International Conference on Wafer Scale Integration is the premier forum for reporting progress in WSI Architecture, Design, Fabrication, and Packaging technologies. Researchers from around the world will describe approaches, technologies, and working WSI devices and systems. Operating hardware will also be exhibited at the conference.

REGISTRATION FORM

Complete and mail this form to:

Michael Yung
Hughes Research Laboratories, RL 69
3011 Malibu Canyon Rd.
Malibu, California 90265
(213)317-5657 FAX: (213)317-5484

Please Indicate Payment Method:

Check (), VISA (), M/C (), A/E ()
Make check payable to: IEEE 1992 ICWSI
Card Number _____

Circle appropriate rate:

Advance Registration (Postmarked by Jan. 8, 1992)
Conference Daily Rate (Circle) **W Th F**
Late/On-Site Registration (After Jan. 8, 1992)
Additional Proceedings are available @ \$40.00 each.

IEEE MEMBER	NON-MEMBER	FULL-TIME STUDENT
\$225.00	\$280.00	\$90.00
\$135.00	\$135.00	\$70.00
\$275.00	\$350.00	\$100.00

IEEE Membership Number _____

Name (Dr., Mr., Ms.) _____

Address _____

Phone _____

Fax _____

Speaker? _____ Session Chair? _____ Committee Member? _____

Call the Fairmont Hotel directly at
(800) 527-4727 to make your room
reservations at the special rate of
\$130.00 per day (single or double).
Indicate that you are attending the
Wafer Scale Integration Conference.



IEEE COMPUTER SOCIETY



THE INSTITUTE OF ELECTRICAL AND
ELECTRONICS ENGINEERS, INC.

Multimedia Conferencing in the Etherphone Environment

Harrick M. Vin, Polle T. Zellweger, Daniel C. Swinehart, and P. Venkat Rangan
Xerox Palo Alto Research Center

Traditionally, voice, video, and data communications have been handled by different communication networks. The public telephone network has been the primary voice carrier. Cable TV has transmitted video. Specialized computer networks have handled data. This historical separation is due to fundamental differences in these media. Audio and video characteristics — such as sensitivity to delay, high bandwidth requirements, and tolerance of high error rates — contrast markedly with the requirements of data. Hence, transmission of audio and video has typically involved circuit switching of analog signals, whereas data has been transmitted through digital packet-switched networks.

Although these media could not be integrated physically, various efforts were made to integrate them logically. One such example is the Etherphone system,^{1,2} developed several years ago at the Xerox Palo Alto Research Center. The original system's primary goal was to integrate the user's view of telephony, other audio applications, and conventional workstation applications. It used Ethernet for audio transmission and control, but separated the two functions, transmitting audio packets over a separate Ethernet via specialized audio hardware. Separate audio hardware improved the system's reliability, performance, and flexibility, and was largely a response to the limited performance of workstations of the time.

The Etherphone system was recently extended to incorporate video. Because networks were not available to carry high-quality digital video, we continued the earlier approach of logical integration in lieu of physical integration. Analog techniques were employed for transmission and switching, while the Etherphone's original flexible control methodology, or *connection management architecture*, sustained the integrated user view.

Limitations of specialized hardware and software prevented full physical integration and further functional extensions of the Etherphone system. However, advances in communication and computer technologies now offer high bandwidth at modest cost and high-performance workstations with multimedia capabilities. For example, Sun Sparcstations include hardware for telephone-quality audio at 64 kilobits per second, and Next workstations implement CD-quality audio at 1.44

The latest extension of the Etherphone project creates a powerful conferencing system that lets users control their participation in multiple conferences across multimedia networks.

megabits per second. Workstations supporting limited-quality motion video are also beginning to appear.

This article describes the latest extension of the Etherphone system. This extension replaces the earlier specialized audio components with Sparcstation-based audio, while retaining the newer video capabilities. We chose to integrate computing with audio processing on the same hardware for three reasons:

- to make Etherphone features available to more users by employing a widely available platform;
- to ease the development of new features by using powerful development environments; and
- to obtain sufficient computing power for additional experiments, such as mixing arbitrary numbers of audio streams or providing independent volume control for each stream.

We have used the hardware enhancements to provide the following new features:

- Users can participate fully in one conference while listening passively to others, providing a flexible means of interacting in a collaborative but distributed work environment.
- A user can establish a multimedia conversation or conference in which both

Evolution of the Etherphone environment

The Etherphone system provides a flexible basis for investigating the management of telephone and video connections and of stored media within a distributed computing environment. These objectives, rather than the exploration of any particular form of media transport (transmission and switch-

ing), have determined the design approach. The goal has been to provide basic multimedia capabilities for use by a variety of workstation applications.

Figure A shows the system architecture at three stages of growth: the original Etherphone system, the Macaw extension, and the Phoenix extension. We will

describe the hardware and user-level capabilities of each stage, then explain the underlying system design that allowed these extensions to plug in with little central change.

Hardware and user capabilities.
The original Etherphone system was

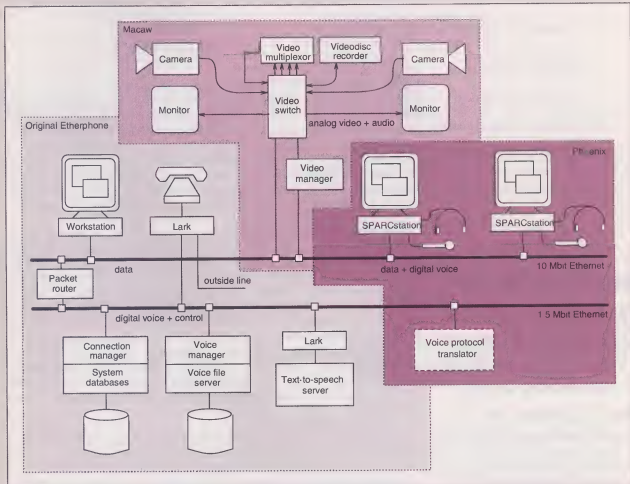


Figure A. Evolution of the Etherphone system architecture.

full and passive conferencing modes are available in audio, video, or both.

This article emphasizes the software mechanisms that support these new features: first, a Sparcstation facility called Phoenix that extends the Etherphone software architecture to permit more flexible conferencing and to control Sparcstation-based Ethernet audio transmission, and second, the integra-

tion of the Phoenix capabilities with Macaw, the earlier video extensions. We also describe our multicast packet protocol for audio transmission, which reimplements and extends the earlier special-purpose protocols, adding per-channel volume control and full support for the extended conferencing modes.

The sidebar entitled "Evolution of the Etherphone environment" describes

the genesis and growth of the Etherphone system and its underlying software architecture.

Before proceeding, we want to highlight other recent efforts to investigate the mechanisms for managing and controlling person-to-person conversations and shared text-oriented workspaces. Lantz² proposes a text/graphics conferencing architecture whose goal is to fit into existing systems with minimal im-

constructed in the early 1980s. The custom-designed Etherphone hardware (known as a Lark) consists of a telephone/speakerphone instrument that includes a microcomputer, encryption hardware, and an Ethernet controller. Larks digitize, packetize, and encrypt telephone-quality voice (64 kilobits per second, with silence suppression), and send the voice packets to each other directly over an Ethernet. Each Lark also has a connection to a standard telephone line. Central services include a connection manager server and a voice file server, both running on a dedicated workstation, and a commercial speech synthesizer configured as a text-to-speech server. A voice manager controls the voice file server and supports sharing and editing of recorded voice via database functions. From a workstation, a Lark user can place and receive telephone calls, maintain private telephone directories, and manage a database of voice messages. An annotation package allows voice to be added to text documents (including electronic mail) and provides a simple direct-manipulation interface for editing voice. In addition, the system can "speak" text under user or program control via the text-to-speech server.

The Macaw extension replaces the audio-only system with a video/audio capability by connecting the Etherphone system to the Media Space, a preexisting network-based control system for Palo Alto Research Center's analog video/audio infrastructure. An analog video switch, controlled by a network server over an RS-232 link, routes video signals from any camera to any set of monitors. An analog video multiplexor, such as a four-way screen splitter, combines multiple video streams into a common view. A commercial videodisc recorder replaces the voice file server for video storage and playback; the video manager is a software database package that supports sharing and editing of recorded video. The original Etherphone connection manager continues to control and coordinate the extended system. Macaw users can place and receive

analog video/audio phone calls from their workstations. The voice annotation and editing package was extended to permit video annotation and editing as well.

The Phoenix extension replaces the specialized Lark telephones with the audio capabilities built into each Sparcstation, which can digitize audio signals at 64 kilobits per second. A Sparcstation may also have associated camera and video monitors, controlled by integrated Macaw software. Again, the resulting system is controlled and coordinated by the Etherphone connection manager, which has been modified slightly to permit the conferencing extensions. The Phoenix audio protocol is sufficiently compatible with the earlier Lark-based protocol that it would be easy to provide a protocol translator permitting Sparcstations and Larks to exchange voice. Phoenix capabilities include flexible multimedia conferencing and simultaneous participation in multiple conversations.

The Etherphone connection manager.

The Lark-based system used Ethernet packet communication for voice transport. However, anticipating the need for other transports as new technologies developed, we designed the system around a transport-independent connection manager. The connection manager provides the basic support for negotiation of call states among the participants in each call. It provides a common notion of state transitions representing the various phases of signaling, connection establishment and holding, and termination. It supports a common approach to reaching parties by name or number. It provides a notification service used to inform each participant of significant actions taken by the others. Finally, it provides registration and notification facilities that permit specialized agents to provide services such as audio recording, audio playback, or voice synthesis.

To achieve transport independence, the components responsible for maintaining a model of the current call state are separated

from the components responsible for implementing and controlling the individual media connections. Within the connection manager are objects called *parties* that know nothing of the details of media, transport, or other local aspects, and therefore abstractly represent each telephone, user window, video terminal, etc., involved in a conversation. Associated with each party is an object called an *agent*, which acts as an intermediary between the connection manager and the specific implementation, converting the individual aspects of a particular transport or of a particular workstation user interface to a standard set of state-management interactions. Agents register dynamically with the connection manager, communicating with it using a remote procedure call protocol, and may thus reside anywhere within the network. Agents retain whatever state is necessary to manage both their participation in conversations and their particular physical configurations.

Two classes of agents are defined. *Transport agents* are associated with actual media terminals such as telephones. They are responsible for control interactions with the connection manager, management of the actual media, and response to local user interactions such as telephone pushbuttons. *Control agents* represent user interfaces such as workstation windows and are responsible only for connection control and user interface support. A new transport is added by creating a corresponding transport agent to represent it. Unless new functionality is also being provided, it is unnecessary to modify the connection manager or the control agent representing the workstation interface. We expect to extend the existing agent set with additional agents in the future, as new networking methods become available to provide superior transports.

pact. Sarin and Greif⁴ have also studied conferencing architectures for text and graphics only. Forsdick et al. at Bolt, Baranek, and Newman,⁵ Ahuja et al. at AT&T Bell Laboratories,⁶ and Aguilar et al. at SRI⁷ propose architectures for person-to-person audio and/or video conferencing. Casner et al. at the University of Southern California's Information Sciences Institute⁸ have developed an audio/video conferencing system integrating all media transport on the same wide-area packet network. Their research has focused on networking and protocol issues of conferencing among geographically distributed sites, rather than on flexible capabilities for user participation in conversations.

Angebrannt et al.⁹ provide a client-server architecture analogous to X Windows for integrating audio and telephony into a graphics workstation. Its emphasis is on lower level audio-resource management rather than on rich conferencing capabilities. The PX system,¹⁰ developed at Bell Northern Research's Computing Research Laboratory, is perhaps most like the Etherphone. Personal workstations cooperate with a flexible control program for a digital switch to explore workstation-controlled telephony and capabilities for editing and distributing recorded voice messages.

Phoenix conferencing capabilities

Voice and video differ fundamentally from data in their semantics and usage, and hence require different paradigms for access. In our system, media interactions among participants take the form of *conversations*, by which we mean any connection between two or more users or between users and multimedia services. A conference is a conversation involving more than two participants. The Etherphone/Macaw system, facilitated by the connection manager, permits a wide range of user actions in the management of conversations:

- *Simple conversations.* A user can make ordinary telephone calls or video phone calls, including multiparty conference calls.
- *Conversations with software services.* A user can establish a conversation with a radio service; monitor a remote

meeting; record, play, and edit recorded audio or video; or cause selected text to be spoken by a voice synthesizer server.

- *Call filtering.* Users can filter incoming calls based on urgency, subject matter, etc.

- *Dynamic membership control.* Users can dynamically join or leave conversations.

- *Call forwarding and visiting.* People do not always remain in their offices. Call forwarding automatically forwards calls to the workstation where a user is logged in. Visiting causes calls to ring simultaneously at the user's original site and at a temporary site, which is identified either by explicit command or by locator devices.

- *Background conversations.* The system can automatically place background calls, such as meeting broadcasts, on hold to accept higher priority incoming calls.

The Phoenix extensions provide the following additional features:

- *Full audio/video conferencing.* Macaw video is combined with Phoenix audio to provide full audio/video conferencing. The analog audio in the earlier Macaw system could not automatically support voice conferencing; it required manual control of an audio mixer.

- *Best-effort conferencing.* Users can participate in conferences using only audio, only video, or both. Each user can choose a different mix.

- *Multiple simultaneous conversations.* Each user may now participate in more than one conversation simultaneously. For example, while participating in a conference, a user may converse with someone else (termed a *side chat*) without disturbing the rest of the conference.

The accompanying sidebar, entitled "Phoenix user interface," demonstrates how to invoke these features.

Multimedia conversations. The lack of audio-mixing hardware limited the Macaw analog audio/video conferencing. The Phoenix extensions combine analog video with digital audio (that included more flexible software mixing), and extend connection management to support conversations in a combination of media.

The ability to create conversations in multiple media gives rise to the concept of best-effort conversations. Two main approaches could be used in creating best-effort conversations:

- *Use the media supported by every participant.* The system would create a conversation in the intersection of the media set that each participant could support. The system would determine the media set dynamically based on hardware availability, forcing all participants to use the same media. If even one could not support video, an audio-only conversation would result.

- *Use the desired media.* This more general model permits asymmetric participation in conversations. For instance, a lecture may be broadcast in the audio-visual domain, but students can participate by viewing only the slides (video only), listening only to the lecture (audio only), or doing both. We have taken this more lenient approach due to its generality and flexibility.

When a participant engages in multiple conversations involving audio, the system digitally mixes all the audio streams received for each conversation. Similarly, if the conversations involve video, the system displays the video streams received for each conversation simultaneously, using a video image multiplexor that places up to four video images on the screen.

Multiple simultaneous conversations.

In real-life situations, a person generally participates in only one interaction at a time, but can passively monitor many other things while doing so. To represent this mode of interaction, the Phoenix extensions permit users to participate in multiple conversations simultaneously, providing a convenient environment for collaborations. For example, during a multiperson meeting, participants forming a subcommittee may convene privately and later report to the meeting (see Figure 1a), or a participant may consult privately with an outside party (see Figure 1b). Participants in either of these side chats continue to monitor the original meeting. The Etherphone system originally provided only simple two-way conferencing and special one-way support for monitoring meetings.

There are two models for supporting these extensions. The first permits ac-

Phoenix user interface

Figure B shows a view of Harrick Vin's telephone management window. Vin is calling his coauthors — Swinehart, PolleZ (who is currently on hold), and Venkat — about this article. During the call, he has placed a side chat call to his secretary, Fear, to resolve a timely question. Vin continues to listen to Swinehart and Venkat converse while he talks with Fear.

To place a conference call, a caller

types a list of recipients in the dialing area or selects several recipients from the conversation log. If desired, the caller can fill in a subject or cycle among several choices for the Priority and Media fields. The control menu includes conferencing features that let callers terminate ringing for recipients who have not answered, add new recipients to an ongoing call, or place side chat calls (which automatically place the old conversation

in a Recv-only state). Additional accelerators and features, such as manually toggling the Recv-only condition to mute the conversation, are available via mouse clicks on conversation log entries. The conversation log distinguishes between Active and Recv-only states for the current user only. To protect privacy, conversation logs of other participants display both states as "active."

Finch 7.0							
Phone	Answer	Disconnect	SpeakText	StopSpeech	Directory	Comment	
ConfEstablished		Invite	SideChat	SetPriority			
Called Party:		Fear.pa			Calling Party:		Pier.pa
Subject:		release forms?			Priority:		urgent
Media:		Audio					
June 25, 1991 10:36:42 am PDT							
42: Placing sidechat call to Fear.pa							
42: Subject: release forms?							
24 Jun 91	4:54:41pm	completed	00:05:23	conference call from Swinehart.pa			
	Subject: Phoenix meeting		Priority: normal	Media: AudioVisual			
		idle	Swinehart.pa				
		idle	Vin.pa				
		idle	PolleZ.pa				
25 Jun 91	9:13:37am	abandoned	00:00:23	from Want.pa			
25 Jun 91	9:48:49am	busy	00:00:14	to Terry.pa			
25 Jun 91	10:27:12am	active	00:10:15	conference call from Vin.pa			
	Subject: IEEE paper		Priority: important	Media: BestEffort			
		receive only	Vin.pa				
		active	Swinehart.pa				
		on hold	PolleZ.pa				
		active	Venkat.pa				
25 Jun 91	10:36:42am	active	00:00:45	to Fear.pa			
	Subject: release forms?		Priority: urgent	Media: Audio			

Figure B. Workstation telephone management window. The Phoenix user interface includes (from top to bottom) a control menu, a dialing area, a system status area, and a conversation log. The log shows a permanent history of this user's calls, including the current state of each conference participant. In this example, the last two calls are still in progress.

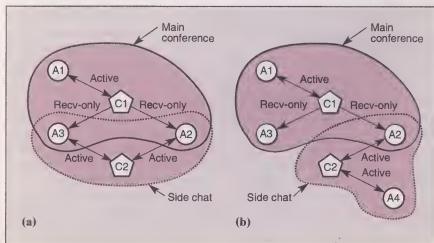


Figure 1. Side chat within a conference: (a) Users A₂ and A₃, participating in C₁, are engaged in a side chat C₂ while also listening to the main conference C₁; (b) A₂ is conversing with A₃ (in C₂) while also listening to C₁. None of the other participants of C₁ can hear the side chat.

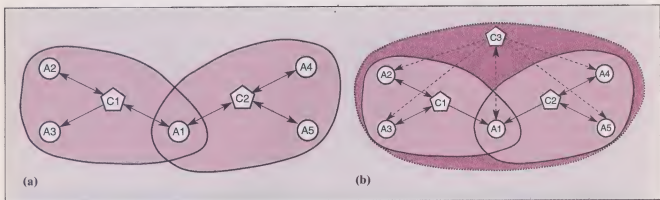


Figure 2. Active participation in multiple conferences. In (a), a participant is sending to more than one conversation simultaneously (to-many, from-many). In (b), the example in (a) is reduced to a to-one, from-many case by introducing dummy conversation C_3 . Arrows show the direction of transmission in each conversation.

tive participation in multiple conversations, yielding a (*transmit*)-to-many, (*receive*)-from-many model of collaboration. The second permits active participation in only one conversation with passive listening to any others (a to-one, from-many model). Phoenix conferencing employs the to-one, from-many model because it captures most collaborative scenarios and is straightforward to implement. It is also functionally complete, since to-many, from-many scenarios can be reduced to a combination of to-one, from-many conversations, as shown in Figure 2. In fact, both these models can also be implemented by a composition of one-

to-many simplex communication channels, such as those described by Crutcher and Grinham.¹¹

Software architecture

This section describes the major software components and activities of the Phoenix version of the Etherphone system. Figure 3 shows the following components:

- *The connection manager.* Cooperates with transport agents and control agents to control the progress of multimedia conversations. (The earlier side-

bar on the evolution of Etherphone discusses the underlying system architecture.)

- *The Finch control agent.* A user workstation module that provides a window interface for placing, joining, and terminating conversations. It also maintains a persistent conversation log.

- *The media transport agent.* Controls media transmission. Two modules provide its functionality: PhoenixAudio, which implements the multicast audio packet protocol, and MacawVideo, which communicates with the video server to control the switching of analog video and audio. A top-level software module works with the connection manager to coordinate the activities of the two media-specific modules.

- *The multicast server.* Helps implement the Phoenix audio packet protocol.

- *The video server.* Connects MacawVideo to the analog video switching infrastructure.

The multicast server and video server are examples of transport-dependent services. These services are implemented as separate servers rather than as components of the individual agents for two reasons. First, each manages contention for a shared resource. Second, each can thus be used for applications other than conferencing.

Agents and servers communicate using a remote procedure call protocol. Several RPC implementations are available,

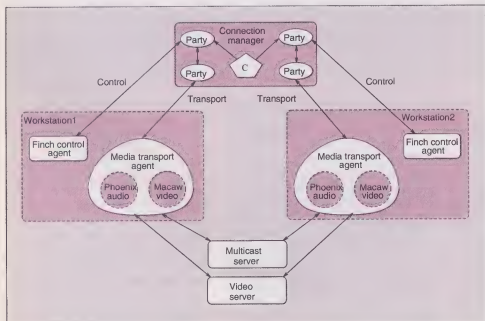


Figure 3. Phoenix software architecture.

but all of the agents described in this article use Sun RPC.

Transport-dependent servers. The multicast server supports PhoenixAudio. When there are more than two participants, internetwork packet-multicasting techniques are used to avoid transmitting more copies of each audio packet than necessary. The multicast server has two management functions to centrally manage the address space for the multicast groups supporting these conferences: `GetMulticastAddress()` creates a new multicast group; `FreeMulticastAddress()` frees one.

The video server is the interface between MacawVideo and the Media Space, PARC's existing analog video/audio infrastructure. The video server provides a channel allocation function for analog media and translates MacawVideo requests into sequences of simple commands to control the video switch and related peripherals. `ReserveMultiplexor()` reserves a video multiplexor for a conversation; `FreeMultiplexor()` makes one available for reuse. `ConnectCamera()` routes video originating from a specified camera or other video source to a multiplexor input port. Lastly, `ConnectMonitor()` routes video from a multiplexor output to a specified destination. The current implementation always uses the four-way video multiplexor, although the video server can support direct, full-screen connections.

Connection manager concepts. Connection management includes the following basic principles:

- *Negotiated conversation establishment.* Setting up a conversation is a negotiation between the caller's agent and the agents of the other participants, whom we call *recipients*.

- *Autonomous participation in conversations.* Every agent is autonomous in that each one can independently decide whether to participate in a conversation and when to leave it.

The connection manager provides functions to create conversations and to control the progression of agents through conversations. An agent can request the creation of a conversation by calling the function `CreateConversation()`, and can then invite other participants to join by calling `Alert()`. (The recipient party enters the state *Notified*, after which its

own agent controls all further activity.) An agent informs the connection manager about its state changes using `Advance()`, and the connection manager uses `ProgressReport()` to report these transitions to all other agents. Similar functions allow agents to register specialized services (for example, speech synthesis) and to be notified when these services generate distinguished events (for example, the completion of one synthesized utterance or the beginning of the next one).

When an agent A wishes to change its state in a conversation, it issues an *Advance* request to the connection manager. The connection manager verifies that the transition is permissible and updates the official version of the conversation state to the new value. Subsequently, agent A takes any necessary local actions to meet the requirements of the new state, while the connection manager concurrently sends *Progress-Reports* to the other participants. Other agents can respond by merely noting the state change, or they can repeat the process by issuing their own *Advance* requests. The sidebar entitled "Telephone conversation model" describes the meanings of the states that agents use to control their progress through conversations.

Functionality of the media transport agent. We now describe the media-specific actions performed by PhoenixAudio and MacawVideo to properly maintain the state of the media transport agent in a conversation.

- *Idle → Initiating.* When the caller's Finch control agent makes this transition, the caller's media agent responds. The media agent's PhoenixAudio takes no action, but its MacawVideo makes two requests of the video server: first, to route the caller's camera to one of the input ports of a video multiplexor assigned to this conversation and, second, to route the output of the multiplexor to the caller's own monitor. The appearance of the caller's camera image on the monitor is a video equivalent of a "dial tone."

- *Idle → Notified.* In the *Notified* state, the agents perform call-filtering decisions. No transport-specific actions are taken by PhoenixAudio or MacawVideo.

- *Notified → Ringing.* PhoenixAudio starts playing a personalized ring tune through the recipient's workstation

speaker. MacawVideo connects the caller's camera to the recipient's monitor in a one-way connection that is the video equivalent of a ringing telephone.

- *Initiating → Ringback.* If no other audio conversation exists, the caller's PhoenixAudio also starts playing the recipient's ring tune. MacawVideo takes no action. Thus, a video-only user will receive only limited feedback, in the Finch workstation window, during call setup.

- *Ringing → Active or Ringback → Active.* The media agent's PhoenixAudio directs the audio protocol implementation to begin sending and receiving packets for the connection or to mix the new party's audio into any ongoing conversation. If the transition is from *Ringing*, then MacawVideo requests the video server to establish a video conversation by routing its camera to the multiplexor assigned to this conversation. If the transition is from *Ringback*, MacawVideo takes no action. If the user is already participating in a video conversation, an additional per-user multiplexor could simultaneously display video streams being received from all the conversations. Figure 4 (on the next page) illustrates how two users establish a video conversation.

- *Active → Recv-only.* In the *Recv-only* state, PhoenixAudio stops transmitting audio packets to the multicast group associated with the conversation, and MacawVideo requests the video server to disconnect the user's camera from the multiplexor associated with the conversation.

- *Recv-only → Active.* PhoenixAudio resumes transmitting audio packets to the associated multicast group, and MacawVideo requests the video server to route the user's camera to the next available input port of the multiplexor assigned to the conversation.

- *Active → Inactive:* This participant is putting the conversation "on hold." PhoenixAudio terminates voice transmission and reception, and MacawVideo directs the video server to disconnect all connections to or from the assigned multiplexor. The agents maintain call information, such as the multicast group address, for use in rejoining the active conversation later.

- *Any state → Idle.* PhoenixAudio and MacawVideo terminate all connections and transmissions, then delete all information regarding the conversation from their information bases.

Etherphone conversation model

Each agent participates in negotiations to establish conversations. In this process, the agent progresses through the sequence of states shown in Figure C. The meaning of each state in this transition diagram is as follows:

•**Error.** An audible error notification, such as a busy signal, is being given to the user. The user must take explicit action to terminate this condition.

•**Idle.** An agent is no longer participating or has never participated in this conversation.

•**Initiating.** The caller's agent enters this state in the process of alerting a recipient's agent of an attempt to establish a conversation.

•**Notified.** A recipient's agent is being alerted about an attempt to establish a conversation. This agent then uses call-filtering information specified by the user to determine its next state. For example, if a user has declared calls from a caller as "hotline" calls, then when alerted by that caller, the user's agent directly enters the Active state. Otherwise, the recipient's agent normally enters the Ringing state, but

might instead decide, based on other user specifications, to reject the call by returning to Idle.

•**Ringing.** When receiving an attempt to establish a conversation, a recipient's agent enters the Ringing state from the Notified state, indicating that an attempt is being made to alert a user.

•**Ringback.** The caller's agent enters Ringback upon learning that the recipient's agent has entered Ringing, indicating to the caller that the recipient is being alerted.

•**Active.** This is the receive-transmit state. All Active agents have bidirectional channels established among them. The recipient's agent enters Active from Ringing as soon as the recipient or a programmed agent has indicated an intent to do so; the caller's agent enters Active after learning that the recipient has answered.

•**Inactive.** This is the "hold" state. An agent enters inactive to place a conversation on hold. Thereafter, it transmits or receives no information for this conversation. In multiparty calls, the agents of other Active participants may continue to converse.

•**Recv-only.** In this state, an agent does not transmit in any media to a conversation. However, the agent continues to receive information from the other participants.

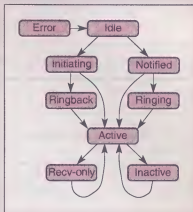


Figure C. State diagram for conversation model. In addition to the transitions shown, there is an implicit path from all the other states to the Idle and Error states. Initiating and Notified are parallel states

from the previous packet also determines the anti-jitter delay that is introduced at the beginning of each talk spurt. This delay provides the buffering necessary to accommodate network delay variations.

The receiver coordinates its local time with the receipt of the first packet of a talk spurt and uses the coordinated real time to predict the scheduled playback time of packets from that sender. The packetNumber and transmissionTime fields in the packet header are used to enforce the ordering of received packets and to correct for differences in the clock rates of sender and receiver. If the sender's clock rate is slower than the receiver's clock rate, then the sender's packets will arrive at the receiver after the receiver's clock has passed its predicted playback time. Rather than dropping all such packets, the receiver periodically resynchronizes with the sender's packets (after receiving a fixed number of late packets) by incrementing the predicted playback time by one packet. Similarly, the protocol detects packets arriving too early. When a packet is received with a scheduled playback time

too far in the future, the receiver synchronizes with the sender's stream by dropping all the buffered packets and then playing this "too early" packet. Both procedures result in noticeable click-like interruptions, which are tolerable if they are not too frequent. The sending and receiving clocks must agree well enough for the typical talk spurt to be shorter than the time it takes the receive buffers to underflow or over-

flow. It is easy to achieve this accuracy with current crystals.

Performance evaluation

The Phoenix extensions have been in operation at Xerox PARC for several months, and the system's performance

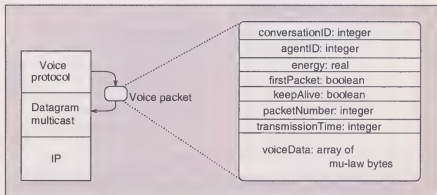


Figure 5. Protocol stack and packet format.

Table 1. Typical performance measurements for connection management operations.

Connection Management Actions	Time (in milliseconds)		
	Audio Domain	Video Domain	Audio +Video
Delay between caller initiating a conversation and recipient being alerted	100	172	172
Delay between caller initiating a conversation and receiving ringback	180	252	252
Delay between recipient clicking Answer and caller going to active state	100	172	172

Table 2. Performance measurements for voice protocol.

Factors	Time (in milliseconds)
End-to-end delay for voice samples	50
Processing delay for each packet at the sender	4
Time required to mix and adjust volume of two audio packets	3

has been within the tolerable limits of commercial telephones. This section describes performance measurements to validate this claim.

Table 1 summarizes some notable state transition times for our conferencing system. Note that the transition times of an agent with respect to a conversation are dependent on the media, since different techniques and hardware are used to perform state-specific actions in the audio and video domains. In evaluating these performance measurements, be aware that a Sun-to-Sun RPC averages 10 milliseconds, the hardware response time of the video switch controlled by a Sun workstation averages 62 milliseconds, connection manager operations such as Advance take about 20 milliseconds, and a ProgressReport reaches other agents in about 80 milliseconds. Multiple processes in the connection manager allow sending ProgressReports to multiple agents in parallel.

In the audio domain, delays result purely from the overhead of communicating state transition information. The

first and third rows in Table 1 sum the times required for an agent to receive a ProgressReport and to Advance its state in response. The second row also includes the time required for a ProgressReport, resulting from the recipient's Advance, to reach the caller. Establishing a conversation involving video requires communication with the video server (hence, with the video switch controller), which causes additional delays.

Table 2 summarizes the most salient performance measurements of the voice protocol. The end-to-end delay is the sum of the packetization delay (20 milliseconds), the transmission delay (including sender processing delay, packet transmission, reception and receiver processing time), and the anti-jitter delay (10 milliseconds). The sender's processing overhead includes the voice energy computation, silence detection, and other functions performed for smooth silence-to-sound transition.

Future work

We have identified areas for further work, including access control, a lightweight connection management model, and media synchronization.

The Etherphone connection management should be enhanced to support different types of conversations. For example, in classroom conversations, each user may have different participation (or access) rights. These rights determine the type of media transmission

channels that must be created for that conversation. For example, if a participant only has permission to receive, then it is necessary to create a simplex communication channel. It is possible to simulate a simplex channel by discarding packets received from a participant with no transmission rights. However, to actively prevent such a participant from sending requires lower level access control (for example, multicast group membership controls).

We believe that future applications such as video browsing will require a lightweight approach to establishing and managing connections. An interesting research issue is the integration of such lightweight applications with our current, more deliberate, negotiation-based models.

Since our system transmits digital audio and analog video, the corresponding delays can vary. The current implementation does not yet provide explicit means to synchronize these media streams. Media synchronization is receiving attention from several researchers, among them Nicolaou.¹²

The Phoenix extensions to the Etherphone system are a step toward integrated multimedia conferencing. They are built on Sparcstations, widely available platforms that can support powerful user interfaces, direct digital conversion and transmission of audio, and advanced telephone-like control functions. Video is transmitted over a separate cable TV network. The overall system achieves physical integration of audio and logical integration of video.

The enhanced conferencing paradigm extends Etherphone telephone conferencing to let users participate in multiple conversations employing multiple media. The Phoenix extensions support a collaboration model in which users can participate actively in one conversation while participating passively in others. The model permits ready extension to active participation in multiple conversations. In addition, the Phoenix control methods permit asymmetric conversation participation (with respect to media used).

We believe that the experience gained from building the Macaw and Phoenix extensions will prove valuable in the future design of completely digital multimedia conferencing systems. ■

Acknowledgments

P. Venkat Rangan was supported by NSF Research Initiation Award No. NCR-9009757.

We thank the referees and Eve Schooler for valuable comments on this article. We also thank Molly Mackinlay for her patience.

References

1. D.C. Swinehart, "The Connection Architecture of the Etherphone System," Tech. Report CSL 91-8, Xerox PARC, Palo Alto, Calif., 1991.
2. P.T. Zellweger, D.B. Terry, and D.C. Swinehart, "An Overview of the Etherphone System and its Applications," *Proc. Second IEEE Conf. Computer Workstations*, IEEE, New York, 1988, pp. 160-168.
3. K.A. Lantz, "An Experiment in Integrated Multimedia Conferencing," *Proc. ACM Conf. Computer-Supported Cooperative Work*, ACM, New York, 1986, pp. 267-275.
4. S. Sarin and I. Greif, "Computer-Based Real-Time Conferencing Systems," *Comput.*, Vol. 18, No. 10, Oct. 1985, pp. 33-45.
5. H.C. Forsdick, "Explorations in Real-Time Multimedia Conferencing," *Proc. Second Int'l Symp. Computer Message Systems*, IFIP, Geneva, Sept. 1985, pp. 331-347.
6. S.R. Ahuja, J.R. Ensor, and D.N. Horn, "The Rapport Multimedia Conferencing System," *Proc. ACM Conf. Office Information Systems*, ACM, New York, 1988, pp. 1-8.
7. L. Aguilar et al., "Architecture for a Multimedia Teleconferencing System," *Proc. ACM SIGCOMM 86 Symp. Comm. Architectures and Protocols*, ACM, New York, 1986, pp. 126-136.
8. S. Casner, K. Seo, W. Edmond, and C. Topolcic, "N-Way Conferencing with Packet Video," Tech Report ISI/RS-90-252, USC/Information Sciences Institute, Marina del Rey, Calif., 1990.
9. S. Angebrannndt et al., "Integrating Audio and Telephony in a Distributed Workstation Environment," *Proc. Summer 1991 Usenix Conf.*, 1991, pp. 419-436.
10. R. Kamel, K. Emami, and R. Eckert, "PX: Supporting Voice in Workstations," *Comput.*, Vol. 23, No. 8, Aug. 1990, pp. 73-80.

11. L. Crutcher and J. Grinham, "Connection Management for an Integrated-Services Network and its Application to the Multimedia Communications of a Distributed Team," Tech. Report TR-90-062, Int'l Computer Sciences Institute, Berkeley, Calif., 1990.

12. C. Nicolaou, "An Architecture for Real-Time Multimedia Communication Systems," *IEEE J. Selected Areas of Comm.*, IEEE, New York, Vol. 8, No. 3, Apr. 1990, pp. 391-400.



Harriek M. Vin is a doctoral candidate in the Department of Computer Science and Engineering at the University of California, San Diego. His research interests are multimedia computer systems and ultra high speed networking with emphasis on conferencing and storage architectures for digital video and audio. He is a recipient of an IBM Doctoral Fellowship and an NCR Innovation Award, and was a research intern at Xerox PARC in 1990.

Vin received his BTech in computer science and engineering from the Indian Institute of Technology, Bombay, India, in 1987, and his MS from Colorado State University in 1988. He is a member of the IEEE Computer Society.



Polle T. Zellweger joined Xerox PARC in 1981 and has been active in the Etherphone project since 1985. She is currently manager of the Active Documents Group in the Elec-

tronic Documents Laboratory. Her research interests include human-computer interaction, multimedia systems, and documents that contain dynamic behavior, such as hyperlinks, audio, video, and computations.

Zellweger received a BA in mathematics from the University of Alaska, an MS in computing and information science from the University of New Mexico, and a PhD in computer science from the University of California at Berkeley. She is a member of the IEEE Computer Society.

Daniel C. Swinehart is a principal scientist at Xerox PARC, performing research in high-speed communications and connection-management methodologies in support of future multimedia applications. His research interests have also included distributed systems, graphical user interfaces and editors, programming environments, and laser-printer software architectures. He is the principal architect of the Etherphone system.

Swinehart received his PhD in computer science from Stanford University in 1974. He is a member of the IEEE Computer Society.



P. Venkat Rangan directs the Multimedia Laboratory at the University of California, San Diego, where he is an assistant professor of computer science. He is also a visiting scientist at Xerox PARC. His research interests are multimedia computer systems and networks. Rangan has co-authored more than 15 publications, and has been a recipient of NSF research initiation and NCR innovation awards.

Rangan received his BTech degree from the Indian Institute of Technology, Madras, India, where he was awarded the President of India Gold Medal in 1984. He received his PhD in computer science from the University of California at Berkeley in 1988. He is a member of the IEEE Computer Society.

Readers can write to Harriek Vin and Venkat Rangan at the Dept. of Computer Science and Eng., Mail Code C-0114, UC San Diego, La Jolla, CA 92093. E-mail addresses for the authors are vin@cs@ucsd.edu., Zellweger.parc@xerox.com, Swinehart.parc@xerox.com, venkat@cs@ucsd.edu..

ADVERTISER INDEX

ASAP '92	Cover II
ASPLOS '92	32
CACI Products Company	1
Carleton University	21
COCS '91	66
COMPASS '92	31
DCC '92	109
IEEE Computer Society Membership	97
Industrial Applications of Fuzzy Control and Intelligent Systems Workshop ..	50
OOPSLA '92	Cover III
Parallel and Distributed Processing Symposium	91
Plenum Publishing Corporation	5
Quantum Software Systems Ltd.	Cover IV
Real-Time Systems Symposium	93
SHPCC '92	30
Siemens Corporate Research	22
University of Minnesota	112
Van Nostrand Reinhold	8
Wafer Scale Integration Conference	68
Classified Advertising	98-102

FOR DISPLAY ADVERTISING INFORMATION, CONTACT:

Northwest: D. Rodney Brooks; tel: (415) 905-0260; fax: (415) 896-1512.

Eastern Region: Georgette Boone; tel: (415) 905-0260; fax: (415) 896-1512.

Southwest/Southeast Region: Kenneth Smitley; tel: (415) 905-0260; fax: (415) 896-1512.

Recruitment & Classified Advertising: D. Rodney Brooks; tel: (415) 905-0260; fax: (415) 896-1512.

Director of Sales: Randall L. Stickrod, South Park Media, 544 Second St., Suite 200, San Francisco, California 94107; tel: (415) 905-0260; fax: (415) 896-1512.

For production information, conference, and classified advertising, contact Heidi Rex or Marian Tibayan.

COMPUTER, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, California 90720-1264; tel: (714) 821-8380; fax (714) 821-4010.

PRODUCT INDEX

	RS#	PG#
ACC Microelectronics	120	90
ADAC	135	91
American Neuralogix	124	90
Animated Voice Corp.	45	89
ATI Technologies	136, 137	91
Boxlight Corp.	40	89
Bridgenorth	138	91
Bright Star Technology	46	89
CACI Products Co.	—	1
Canon USA Inc.	21-25	81-85
Concentric Technologies	43	89
CSP Communications	36	88
Du Pont Pixel	139	91
Electrical Engineering Software	121	90
Hydra Systems	140	91
I&J Finar	122	90
IBM	35	88
IMSI	28	87
Linear Technology	123	90
Micropolis Corp.	26	86
Nth Graphics	141	91
Our Business Machines	27	86
Performance Software	44	89
Philips Semiconductors	125	90
Photron and Synesis	142	91
Plenum Publishing Corp.	1	5
Quantum Software Systems Ltd.	3	C.I.V.
R.C. Electronics	143	91
Silicon Graphics	39	88
Silicon Sports	29	87
Sony	42	89
Specialix	144	91
SuperMac Technology	38, 145	88, 91
Texas Instruments	126	90
Truevision Inc.	37	88
Unitrode	127-128	90
Van Nostrand Reinhold	2	8
VTC	129	90
Wavefront Technologies	41	89
Xicor	130	90

COMPUTER

Reader Service Card

October 1991 issue (Card void after April 1992)

Print address below.

Name _____

Title _____

Company _____

Address _____

City _____

State/ZIP _____

Country _____ Phone _____

Reader interest

(Circle what you liked; add comments on the back)

- A** Guest Editors' Introduction
- B** Digital Multimedia Systems
- C** Conceptual Modeling
- D** Automating Explanations
- E** Spatio-Temporal Composition
- F** Continuous Media I/O Server
- G** InternetExpress
- H** Multimedia Conferencing
- I** President's Message
- J** Product Reviews
- K** New Products
- L** IC Announcements
- M** Microsystem Announcements
- N** Update
- O** Computer Society News
- P** Standards
- Q** Call for Papers
- R** Calendar
- S** Book Reviews
- T** Open Channel

Product information

(Circle numbers for products and advertisements on which you want more information)

1	21	41	61	81	101	121	141	161
2	22	42	62	82	102	122	142	162
3	23	43	63	83	103	123	143	163
4	24	44	64	84	104	124	144	164
5	25	45	65	85	105	125	145	165
6	26	46	66	86	106	126	146	166
7	27	47	67	87	107	127	147	167
8	28	48	68	88	108	128	148	168
9	29	49	69	89	109	129	149	169
10	30	50	70	90	110	130	150	170
11	31	51	71	91	111	131	151	171
12	32	52	72	92	112	132	152	172
13	33	53	73	93	113	133	153	173
14	34	54	74	94	114	134	154	174
15	35	55	75	95	115	135	155	175
16	36	56	76	96	116	136	156	176
17	37	57	77	97	117	137	157	177
18	38	58	78	98	118	138	158	178
19	39	59	79	99	119	139	159	179
20	40	60	80	100	120	140	160	180

COMPUTER

Reader Service Card

October 1991 issue (Card void after April 1992)

Print address below.

Name _____

Title _____

Company _____

Address _____

City _____

State/ZIP _____

Country _____ Phone _____

Reader interest

(Circle what you liked; add comments on the back)

- A** Guest Editors' Introduction
- B** Digital Multimedia Systems
- C** Conceptual Modeling
- D** Automating Explanations
- E** Spatio-Temporal Composition
- F** Continuous Media I/O Server
- G** InternetExpress
- H** Multimedia Conferencing
- I** President's Message
- J** Product Reviews
- K** New Products
- L** IC Announcements
- M** Microsystem Announcements
- N** Update
- O** Computer Society News
- P** Standards
- Q** Call for Papers
- R** Calendar
- S** Book Reviews
- T** Open Channel

Product information

(Circle numbers for products and advertisements on which you want more information)

1	21	41	61	81	101	121	141	161
2	22	42	62	82	102	122	142	162
3	23	43	63	83	103	123	143	163
4	24	44	64	84	104	124	144	164
5	25	45	65	85	105	125	145	165
6	26	46	66	86	106	126	146	166
7	27	47	67	87	107	127	147	167
8	28	48	68	88	108	128	148	168
9	29	49	69	89	109	129	149	169
10	30	50	70	90	110	130	150	170
11	31	51	71	91	111	131	151	171
12	32	52	72	92	112	132	152	172
13	33	53	73	93	113	133	153	173
14	34	54	74	94	114	134	154	174
15	35	55	75	95	115	135	155	175
16	36	56	76	96	116	136	156	176
17	37	57	77	97	117	137	157	177
18	38	58	78	98	118	138	158	178
19	39	59	79	99	119	139	159	179
20	40	60	80	100	120	140	160	180

COMPUTER

Reader Service Card

October 1991 issue (Card void after April 1992)

Print address below.

Name _____

Title _____

Company _____

Address _____

City _____

State/ZIP _____

Country _____ Phone _____

Reader interest

(Circle what you liked; add comments on the back)

- A** Guest Editors' Introduction
- B** Digital Multimedia Systems
- C** Conceptual Modeling
- D** Automating Explanations
- E** Spatio-Temporal Composition
- F** Continuous Media I/O Server
- G** InternetExpress
- H** Multimedia Conferencing
- I** President's Message
- J** Product Reviews
- K** New Products
- L** IC Announcements
- M** Microsystem Announcements
- N** Update
- O** Computer Society News
- P** Standards
- Q** Call for Papers
- R** Calendar
- S** Book Reviews
- T** Open Channel

Product information

(Circle numbers for products and advertisements on which you want more information)

1	21	41	61	81	101	121	141	161
2	22	42	62	82	102	122	142	162
3	23	43	63	83	103	123	143	163
4	24	44	64	84	104	124	144	164
5	25	45	65	85	105	125	145	165
6	26	46	66	86	106	126	146	166
7	27	47	67	87	107	127	147	167
8	28	48	68	88	108	128	148	168
9	29	49	69	89	109	129	149	169
10	30	50	70	90	110	130	150	170
11	31	51	71	91	111	131	151	171
12	32	52	72	92	112	132	152	172
13	33	53	73	93	113	133	153	173
14	34	54	74	94	114	134	154	174
15	35	55	75	95	115	135	155	175
16	36	56	76	96	116	136	156	176
17	37	57	77	97	117	137	157	177
18	38	58	78	98	118	138	158	178
19	39	59	79	99	119	139	159	179
20	40	60	80	100	120	140	160	180

Editorial comments

I liked: _____

I disliked: _____

I would like to see: _____

For reader-service inquiries, see other side.

PLACE
POSTAGE
HERE

PO Box is for reader-service cards only.

COMPUTER

Reader Service Inquiries

PO Box 16508

North Hollywood, CA 91615-6508

USA

Editorial comments

I liked: _____

I disliked: _____

I would like to see: _____

For reader-service inquiries, see other side.

PLACE
POSTAGE
HERE

PO Box is for reader-service cards only.

COMPUTER

Reader Service Inquiries

PO Box 16508

North Hollywood, CA 91615-6508

USA

Editorial comments

I liked: _____

I disliked: _____

I would like to see: _____

For reader-service inquiries, see other side.

PLACE
POSTAGE
HERE

PO Box is for reader-service cards only.

COMPUTER

Reader Service Inquiries

PO Box 16508

North Hollywood, CA 91615-6508

USA

PRODUCT REVIEWS

Editor: Richard Eckhouse, University of Massachusetts-Boston
Send review submissions to MOCO Inc., 776A Country Way, N. Scituate, MA 02066; Compmail r. eckhouse; Internet, eckhouse@cs.umb.edu.

Canon on review

It's always intriguing to look at different products and technologies offered by the same vendor. In this issue, reviewers evaluate the Bubble Jet and laser printers from Canon, as well as its latest flat-bed scanner.

The BJ-10e

Those who are constantly racing about, rushing to meet deadlines, composing last-minute proposals, and revising research data at awkward hours are demanding hardware to suit their schedules. The laptop computer was their first purchase, and the Canon BJ-10e Bubble Jet printer should be their second. Measuring $8.5 \times 12.2 \times 1.9$ inches and weighing approximately 4 pounds, Canon's printer is very portable and an ideal companion to a laptop. It's also easy to maintain and delivers professional output.

The BJ-10e is simple to operate. It can use an AC adapter or run on an optional battery pack (list price \$50). The nickel cadmium pack can be charged inside the printer and has a useful life of approximately 40 minutes. The user controls the printer with five push buttons and four status lights: Power, On Line, Paper, and Error.

A set of DIP switches, conveniently located and labeled, lets users customize the printer to any of the six character sets. The 10 settings include normal or high graphics densities, and line feed/carriage return options. As a factory default, the printer emulates an IBM Proprinter X24, but it can be set to emulate a Canon BJ-130e. Although a driver wasn't included with the BJ-10e, the emulations work rather well (a Windows driver is available).

The manual paper feed accepts one sheet at a time, which offers versatility when printing on different materials but can become annoying on long runs. I recommend the \$90 optional automatic sheet feeder for larger printing jobs. Still, for small documents, the single-sheet loading pro-

vides accuracy and allows for greater control with special projects. The form feed/line feed button is supplemented by forward and reverse line switches, which lets you line up paper exactly. You can easily feed envelopes to the printer through a bottom slot, with great-looking results.

A plethora of print modes works with this printer. Modes range from normal to emphasized to double high to double wide. HQ mode provides high-quality printing, while Economy mode saves ink. You can set spacing at 10, 12, or 17 characters per inch, or select proportional spacing.

The printer resembles a typewriter in its typesetting options. Choosing printing modes is actually the only tricky part of operating the BJ-10e. The pitch button determines the number of characters per inch when the printer is off line and selects print modes while it is on line, each according to the number of times the button is pressed. My experiences with this frequently sent me to the manual to determine which mode I was in. The printer does sound a differently pitched beep for each mode, but it takes experience to identify the modes on the basis of sound alone. Another difficulty is that the HQ and Economy modes are selected by pressing Formfeed/Linefeed while the printer is on line. Although it is easier to identify these two modes, you can accidentally change modes instead of advancing a line in printing.

If you don't want to bother with setting modes for the printer, however, the BJ-10e will print nearly any font established in the file. I was amazed to see the printer output exactly what was on the screen. What you see is really what you get. However, if you are not printing out of a word processing package, be forewarned that there is very little — if any — top margin.

Graphics are only fair. Although straight horizontal and vertical lines are very crisp, diagonal lines and curves appear a bit jagged. Also, solid colors tend to streak. Add to this the

traditional problems of an ink printer. Solid black objects, for instance, seem to soak the paper when first printed and leave a magic marker effect on the back of the paper even after it is dry. Also, you must be careful not to smudge the ink when you handle individual pages. You should also avoid some bonds of paper that allow the ink to bleed, giving letters a feathery quality. However, this is noticeable only with the larger print modes. With a horizontal resolution of up to 360 dots per inch, many modes have a typewriter quality; some even rival laser print. Times Roman at 12 cpi in HQ mode, for example, is crisp. Script at 12 cpi has an extraordinarily fine appearance.

The BJ-10e proved simple to maintain. Printing is made possible by a replaceable Bubble Jet cartridge, which appears to be the only printer component that must be replaced. Installation is easy and well documented by the clearly written manual. The cartridge (\$25) has an auto-capping system that engages when the printer is turned off or has remained idle for more than 5 seconds. The system helps protect the cartridge head and also keeps ink from drying out, a good feature that helps to maintain the only part of the printer needing maintenance. There is also a self-print-head-cleaning system activated by pressing simultaneously the FWD ADJ and REV ADJ buttons. This takes only a second and a half to complete and is well worth the effort.

This convenient, low-maintenance printer is more than capable of helping you meet those last-minute deadlines. If you need a compact printer at \$499 (this price is heavily discounted to \$349) that will offer low-hassle with quality results, I recommend the BJ-10e.

Readers may contact Canon USA Inc. at 1 Canon Plaza, Lake Success, NY 11042; (516) 488-6721; or circle Reader Service Number 21.

— Karin Feeney, MOCO Inc.

The Canon BJ-330

The Canon BJ-330 Bubble Jet printer is a versatile wide-carriage machine that outputs high-quality graphics and clear text quickly and quietly. Its 360-dpi resolution and large-paper capacity make it a smart choice for the user with graphics in mind. With 36 possible typeface combinations, the printer also offers a great many text options. This machine prints on nearly anything and does so at high speed with few problems.

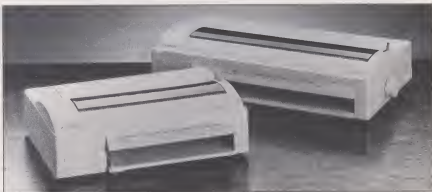
One of the most surprising — and pleasing — strengths of the BJ-330 is its versatility in printing on various materials. The printer adapts well to working with cut-sheet paper, fed manually from the front or automatically from the optional sheet feeder. It works equally well with fan-fold paper and envelopes.

I highly recommend the two-bin automatic sheet feeder (\$150 for the first bin and \$129 for the second). Each bin has a 100-sheet capacity and is readily adjusted for a broad range of paper widths. The dual sheet feeders are great for alternating paper sizes or bonds. When you have plain paper in one bin and letterhead stationery in another, you can easily switch between the two bins, which relieves a lot of paper-loading hassles. Also, stacks of envelopes can be loaded through either bin.

As mentioned, the BJ-330 can work with fan-fold paper, a practical capability unusual to most non-dot matrix printers. Loading is easy and well documented in the user manual. Printer control is similar for both cut-sheet and fan-fold papers, and alternating between the two should not be difficult.

The \$949 BJ-330 is the only printer in its class that can work with a large range of paper sizes. With the sheet feeder, the BJ-330 can handle paper up to 14.3 × 14.3 inches; manually processed paper can measure up to 17 × 23.4 inches, a good feature for those needing poster-size charts and graphics. You do not have to dismantle the paper bins to get the larger output offered by manual paper loading. Even with the sheet feeder installed, you can feed large sheets through the front loading system. That system accepts a minimum of 3.5-inch-wide paper or envelopes. However, it takes a while to insert paper correctly through the front opening; the system jams rather easily.

The BJ-330 has three local fonts — Courier, Prestige, and Gothic — that can be supplemented by optional font cards. Six pitch modes are available:



Canon BJ-300 (left) and BJ-330 Bubble Jet printers.

10, 12, 15, 17, and 20 cpi, and proportional spacing. Settings also exist for High-Quality mode and High-Speed mode. High-Speed mode is very fast indeed, clocking in at 300 cps, with no sacrifice in quality — just less ink. High-Quality mode has a not-too-shabby text speed of 150 cps and lives up to its name in terms of output.

The printer's 64-nozzle ink jet produces fine results. As far as fonts go, there are no jagged edges. One of the few problems, however, comes with those font and pitch modes in which less ink is used: each ink dot is defined, giving the text a somewhat perforated look. Also, on many bonds of paper, the ink bleeds, which gives the otherwise crisp text a fuzzy appearance. I recommend experimenting with different types of paper to find the optimal quality output. Copier paper works rather well, as do most other smooth bonds. Printer detail is particularly enhanced by using cotton paper.

All fonts that I tried with common software packages printed with excellent results. A curious thing happened with one word processing package: Software font settings dominated printer settings. Each time a new pitch mode or printer font was selected, the settings reverted to default (High-Quality Courier at 10 cpi) upon printing. This was corrected through a Lock Out function on the printer that causes it to ignore such computer commands as those governing pitch modes, fonts, and paper sources.

Graphics came out extraordinarily well, with crisp lines, even shading, and defined details. Much depends on the graphics software, of course, but the 360-dpi resolution maintained the quality of the graphics as printed on the screen. There was some striping in solid blacks and the darkest of grays, but all other shades printed beautifully.

The printer ink is mostly smudge-proof. It dries quickly after printing and presents few problems. Even when dampened with a drop of water, text does not smudge beyond recognition.

Printer maintenance could hardly be simpler. The printer has its own head-cleaning system, which takes only seconds to run and clears dust and particles from the print nozzles. An ink-priming function ensures good print quality if there is any danger of ink clogging, even if the computer has not been used in more than six months. (These functions will probably not be needed much.) The only other real maintenance duty requires replacing the \$28 ink cartridge, which reputedly has a life of approximately one million characters in High-Quality mode.

The BJ-330 doesn't have its own driver, but it does have emulations for the IBM Proprinter X24e series and the Epson LQ printers. For those who make their own drivers, or just want to use their own software commands, a programmer manual is included. Printer options are the automatic sheet feeder, a serial interface board, extra font cards, and an expansion RAM card to supplement the 30-Kbyte buffer. The user manual clearly and thoroughly documents them.

This printer is fast and can output professional quality, without a lot of trouble on the part of the user. Canon's BJ-330 offers the paper versatility of a dot matrix, with the speed and resolution of a laser. With plenty of options for output, this is certainly an ink jet printer to consider.

Readers may contact Canon USA Inc. at 1 Canon Plaza, Lake Success, NY 11042; (516) 488-6723; or circle Reader Service Number 22.

— Karin Feeny, MOCO Inc.

PostScript for the LBP-4

In *Computer's* last review of the LBP-4 (May 1990, pp. 90-92), we looked at the basic printer with the optional cassette feeder and memory expansion board. Readers may remember how much we liked the laser, particularly for its versatility and ease of use. Since then we've been waiting for Canon to release its PS-2 Script Card, which transforms the LBP-4 into a true PostScript printer.

The wait was worth it. I received a kit that contains the version 51.4 interpreter from Adobe Systems, along with a floating-point chip and a multilingual user manual. Because I had an earlier version of the RAM extension board, I needed to trade it in for the latest revision that includes an FPA socket (a no-cost upgrade). I removed the four screws that hold the memory board in place, popped the FPA into the new board, put the machine back together, and inserted the PS-2 card into the first cartridge slot. I knew things were different when I switched on the machine and saw a new initialization message (PS Initializing).

Without touching the printer, I now had a PostScript machine with 39 built-in Adobe fonts including Times Roman, Avant Garde, Helvetica (normal, narrow, and condensed), Palatino, Courier, New Century Schoolbook, ITC Bookman, Symbols, and ITC Zapf Dingbats. My old laser took on a new personality that made printing under the Windows environment a real joy.

Because I hadn't received any new software with the upgrade, I was concerned about which drivers to use with Windows. Having already installed the generic PostScript printer driver, I figured "what the heck" and gave it a try. Everything worked perfectly. Documents that I had printed previously with another PostScript printer looked exactly the same (there's a lot to be said for standardization here). And while I had some of this quality available before with the scalable font feature built into the LBP-4, I now had many more fonts and features that were completely compatible with any PostScript printer.

Of course, you can still use the LBP-4 as a Capsl (Canon Page-Specification Language) engine — either by using the menuing system accessible from the front panel or by removing the PS-2 cartridge before powering up the machine. But the real test is to figure out how to send documents to the laser from the PostScript mode. Frankly, I'm not sure about the official method for doing so, but I did dis-

cover an interesting workaround. By using the PSEXEC software that came with a colleague's QMS printer, I sent PostScript and non-PostScript files to the printer with excellent results. But I would still like to be able to load a printer driver under DOS so that I could output simple non-PostScript files to my LBP-4.

The user manual that comes with the \$695 PS-2 card upgrade is sufficiently complete so that users can understand the upgrade installation and the operation of the laser through the use of the control panel. A separate section in the back of the manual is devoted to the basic operation of the ScriptCard, communication modes, server information, system parameters, and PostScript color extensions and language changes. For most, this technical information is more than re-

The Canon LBP-8 Mark IIIR

This laser is for people with serious printing needs. It comes in a hefty package weighing in at 67.1 pounds with a footprint of approximately 18 x 31 inches and furnishes the functions needed by production-oriented publication departments or companies. The Mark IIIR provides the same high-quality output as the LBP-III reviewed here last year, but with important additional features.

For those who missed that May 1990 review, Canon's family of 8-ppm, 300-dpi laser printers starts with the LBP-III, with one paper cassette. Next is the Mark IIIT, a two-cassette version of the base printer. The Mark IIIR — the top of the line — is a two-cassette printer that can print duplex copies (two sides of the page). The Mark III series all contain, not surprisingly, the Canon laser engines; these engines are also used in the industry-standard Hewlett-Packard family of laser printers. Because of OEM agreements, however, Canon's laser printers are not allowed to be HP compatible.

In today's wonderful world of device-independent GUIs, this is a minor problem because Canon supplies a native Windows printer driver that capitalizes on all features of these printers. If these drivers, or Canon's proprietary Capsl, can't deliver the function that you need, an optional PostScript feature is available for all

quired — but not enough for a user to learn all about PostScript.

This laser has been constantly used for the last year and has been a real pleasure to operate. From the high-quality output to the hassle-free operation, this machine has proven itself a real workhorse. With the PS-2 card, it takes on a new personality and offers a significant upgrade. I recommend it highly. And if you haven't purchased the LBP-4 yet, you can get the complete system consisting of a Canon LBP-4 with ScriptCard and a 1-Mbyte additional RAM board for \$2,359 list price.

Readers may contact Canon USA, Inc. at 1 Canon Plaza, Lake Success, NY 11042; (516) 488-6725; or circle Reader Service Number 23.

—Ruth Maulucci, *MOCO Inc.*

printers. If you can't afford the PostScript option, or can't be bothered using Canon's Capsl, the machine can emulate other printers like the Diabolo 630.

The Mark IIIR comes with 1.5 Mbytes of RAM, expandable to 4.5 Mbytes using the optional Expansion RAM Board, a feature that takes 5 minutes to install and test. The machine can select paper from any one of three sources — an upper or lower 200-sheet paper tray, or an optional 50-envelope feeder. The envelope feeder is actually a third "tray" that does not interfere with the other trays. Of course, you can manually feed paper (or even envelopes) into either of the trays. A variety of paper-tray sizes are available as options.

The printer contains eight internal bitmapped fonts that can be magnified, bolded, italicized, etc., from the front panel switches or under computer control. In addition, the printer contains four Swiss fonts, four Dutch fonts, and one Symbol font. Any of Canon's four optional font cartridges can be used to supplement these internal fonts. Alternatively, the current version of the Adobe PostScript option (a cartridge plus a ROM/processor unit) provides 35 fonts.

The printer comes with both serial and parallel interfaces, but the current PostScript version (V. 51.4) does not support AppleTalk or RS-422 commu-

nications. An optional video interface board is also available.

One of the more interesting things about all Canon laser printers is the clever front panel with its feature-selection switches. Besides being controllable from an on-line computer, most of the features can be set and reset interactively by the user. The control panel contains eight switches and an LCD display. The Mark IIIR has more than 35 user-selectable options organized into seven hierarchical categories.

It is very easy to use the front panel switches to select a category, and to navigate through the hierarchy, change a setting, and set it either into temporary or battery-powered memory. The seven categories include a setting that lets you select the language for LCD display (Danish, Dutch, English, French, German, Italian, etc.). Other categories include settings for the interface (serial or parallel options), number of copies or overlays, font selection and manipulation, page layout, and duplex printing options.

The duplex category is really the feature that puts this two-tray printer

in a class beyond conventional (simplex) laser printers. Duplex printing with the Mark IIIR is simple and fascinating. It's simple to make it print on two sides of a page — and it's fascinating to watch the process. Duplex printing starts out like simplex printing. A sheet of paper is fed into the machine, one side is printed as it passes by the print engine, and the paper is fed out to a paper-support device on the other side of the printer. It is then drawn back and fed in reverse past the engine, allowing the other side to be printed. The page is then deposited onto the top side of the printer. The effective page-per-minute rate is diminished through all of this, from 8 to 3.5.

Duplex mode can be used, of course, simply as a paper-conservation technique. It can also be used to easily print multiple copies of professional-looking reports containing pages with two sides of print. To facilitate this, the Mark IIIR lets you set larger margins on the printed pages to allow for left or right binding, in landscape or portrait modes.

You might think that with all these features this printer might be somewhat onerous to set up, use, and maintain. Let me assure you otherwise. The excellent manuals are well laid out, illustrated, and indexed. I have used this printer constantly for months, and except for a few manually misfed pages, I have encountered virtually no problems setting it up, installing options, or maintaining the machine. The few times that the printer jammed, the LCD's concise messages were easy to follow, allowing me to quickly get on with my work. I am convinced that the Mark IIIR is an outstanding top-of-the-line product in Canon's LBP-8 series of laser printers. If you are in search of a high-quality, reliable, workhorse laser, either for use in a network or for professional production output, this printer is for you.

Readers may contact Canon USA Inc. at 1 Canon Plaza, Lake Success, NY 11042; (516) 488-6700; or circle Reader Service Number 24.

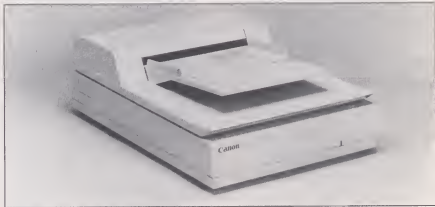
— Sorel Reisman, California State University, Fullerton

The Canon IX-30F

It's not often that you have the chance to review a product that does what it's supposed to do — and does it well. The Canon IX-30F Image Scanner is just such a product. Scanning with or without the optional 20-page feeder tray, this flat-bed scanner performs flawlessly, scanning a page with up to 256 shades of gray at 300 dpi in less than 10 seconds.

The IX-30F that I tested was attached to my 386 PC via Canon's PC/AT interface kit that includes a SCSI adapter board and cable. A similar cable is available for Apple's SE 30 and Mac IIs (without the adapter board). Both PC and Apple kits include Scando, a basic software package that converts scanned images into TIFF, MSP, or PCX file formats.

When I first unpacked the product, I started with some trepidation at the nine-page, four-language, difficult-to-decipher pamphlet that describes unpacking, DIP switch settings (to specify DMA channels, interrupt type, SCSI ID, and I/O address), and installation of the supplied driver. (A 32-page manual in English describing the care and maintenance comes with the drive.) I knew I was in for a challenge trying to make this product work along with my installed fax board, speech recognition board, and SCSI-



Canon IX-30F Image Scanner.

attached CD-ROM player. I wasn't wrong. Three hours later, with a retrospective analysis of the contents of the brochure, I understood everything the brochure said. But the effort had not been wasted.

In addition to Scando (which I did not review), Canon provides a list of scanner software products that support the IX-30F. These include OmniPage/386 (Caere Corp.), ReadRight (OCR Systems Inc.), Gray F/X (Xerox Imaging Systems Inc.), Image-In (Image-In Inc.), Picture Publisher (Astral Development Corp.), and the WordScan products (Calera Recognition Systems Inc.). I performed most of my testing using WordScan Plus, a

really great product for optical character recognition (OCR).

WordScan's installation instructions explicitly direct Canon scanner owners not to install the Canon-supplied driver, but instead to use Calera's driver. This would have been okay except for contradictions regarding DIP switch settings for IRQ and I/O address selection. (Canon's instructions were correct.) WordScan includes a help selection option called the Technical Assistant that examines your config.sys and autoexec.bat files and detects conflicts among paths, device driver parameters, and adapter board switch settings. The Technical Assistant can correctly modify the soft-

ware-related items until they do not conflict with DIP switch settings. Of course, it's your own responsibility to make sure that those are not in conflict with other attached peripherals.

WordScan is very easy to use. There are two stages when using it with the IX-30F: scanning documents and then directing the software to examine the scanned image as either text, image, or both. WordScan's primary IX-30F programmed features let you menu-select scanning brightness and the orientation of the source document, and specify whether the source document is "normal" or from a dot matrix printer. You can also specify whether the source document is text, image, or a combination of both. Presumably, these selections produce optimized hardware settings for the source type.

I rarely had to go to the secondary hardware-adjustment feature menu, and then it was usually to select binary versus a dither pattern for text-versus-image source documents. The secondary menu features let you, under control of the software, incrementally adjust scanning contrast; select either full (binary) image scanning or one of eight dither patterns, dpi resolution (75/150/200/300), and paper size (letter, A4, or legal); and choose the page feeder versus the flatbed.

When using WordScan, you can select Automatic Mode, which simply uses the text/image/both setting that you select before scanning. Or you can select Preview, which lets you, after the image has been scanned, identify areas on the image that should be

processed as text, image, or both. By default, this feature also lets you omit areas for processing. Consequently, you can expedite the processing speed by using Preview mode. WordScan also lets you define templates based on these selections, thereby facilitating the scanning of pages with similar content layout.

After all is said and done, the real question is how well the OCR software works. To be honest, I was amazed at the low error rates that resulted, regardless of the kind of source document used. I tried multi-font, columnar magazine articles, conventional laser-printed material, and dot matrix output. In every case, with only one minor deviation from the scanner's default settings, the scanned text contained very few errors. And the "pop-up verifier" contained enough text-processing functions (including user-definable dictionaries) to let me save error-free documents in many file-text formats. The manuals (and Calera's very friendly and knowledgeable product support staff) really simplify what not too many years ago was a very difficult, if not impossible, OCR process.

One of the best product features is ScanIn!, the macro that installs the software as a selectable item in the Word For Windows' file-option list. By faithfully following (but not really understanding) the installation instructions, I could place my cursor anywhere in a Word For Windows document, select the new Scan/OCR option, have WordScan automatically

scan a document, and place the recognized text directly into my current document. Images are saved as TIFF files that can be inserted using Word For Windows' Insert Picture options. This scenario is part of Microsoft's strategy that recognizes that, for most users, the word processing package is the working desktop. By using WFW's powerful macro language, a host of word processing features (fax, OCR, charting, etc.) can be easily integrated into WFW's extant desktop metaphor.

Although I really liked WordScan, I should mention that — unlike some of its competitors — the program is not trainable. While I have found this feature desirable for documents with unusual kinds of fonts, I have also found these programs unreliable. Neither can WordScan print directly from the package. If you have scanned an image (or used OCR without WFW), you must save the input for import into another paint/draw or word processing program.

Because of WordScan's low error rates and IX-30F's ease of use, I heartily recommend this combination for almost any production-minded OCR environment. And if WordScan's features don't quite suit your requirements, you are sure to find a substitute software package.

Readers may contact Canon USA Inc. at 1 Canon Plaza, Lake Success, NY 11042; (516) 488-6700; or circle Reader Service Number 25.

— Sorel Reisman, California State University, Fullerton

Review notes

1684 PC Pak. When I received this product from Micropolis, I was impressed by how carefully and thoroughly the packaged drive had been put together. Not only does the kit include the Micropolis drive, an Always SCSI interface, and an excellent installation manual, but it was complete down to the rails I needed for installation. A good extra touch was the bundled software that included device drivers for Novell Network and Xenix, should you want to install the drive within a server or as a Unix device. For DOS users there were also Disk Manager, PC-Kwik Power Pak, and PC-Kwik Power Disk. Clearly, this drive can be installed in many different environments, although I only tested it within DOS 5.0.

At 340 Mbytes, the Micropolis is a large-capacity drive for a PC. Yet,

physically, the drive is only half-height, and the controller board is only half-length. So you can put in two SCSI drives and connect them to the controller using the dual connector drive cable that comes with the device. Of course, the controller will also support two floppy drives. Hard-drive installation is extremely easy and is well documented in the installation guide. You just remove your old hard disk and controller, install the new disk and controller, and reable to the new disk (the interface cable is different for the hard drive but not the floppies, so you can use the same cable as before). Getting your machine to recognize the new drive is a bit more complicated to explain than to perform, but here goes.

The first step is to tell your BIOS that you don't have a hard drive. Next

you use the debug program to perform low-level formatting using the SCSI host adapter's BIOS at the formatting routine's location. You next call FDISK to set up the partitions on the drive, and finally you call FORMAT to create a bootable partition with DOS on it. Now you are ready to reload the software that you backed up from your old drive and start using the Micropolis drive.

Two things are immediately noticeable. First, the Micropolis drive is generally faster than your old drive. While I could deduce this from an average track-to-track seek time of 15 milliseconds, what I found was that streaming tape backups took about 20 percent less time and disk-intensive software booted up faster. Actual benchmark results using PC Tools V7 confirmed the average seek time and

Tape backup

The Micropolis drive, being 20 percent faster, gave me DMA problems with my tape backup. I have a separate tape controller board that includes hardware compression. Randomly, the backup would hang up the machine (a 486/33), which had to be rebooted. Turning off concurrent disk and tape operation solved the problem, but at a cost of nearly doubling the backup

time. Changing the tape controller configuration (IRQ and DMAs used) didn't cure the problem, but removing the controller card did. I can still have concurrent tape and disk operation, but backup times increased by 67 percent. Although I thought a later version of the Colorado software might fix the problem, Colorado's technical support says it is in PC's DMA controller.

the higher transfer rate of this drive over the 210-Mbyte IDE drive it replaced. The overall performance composite was 27.7 for the IDE drive and 32.2 for the Micropolis, giving the Micropolis a 16 percent advantage. Comparing the Micropolis to an IBM PS/2 Model 90 (21.0) shows an even bigger difference of 53 percent. Either way, larger capacities can also mean faster speeds!

This drive differed from my old one in that cluster size was doubled (8 versus 4 Kbytes). Although this variance results in more internal slack per file, it was negligible when I compared the two (about 4 percent more for the Micropolis). Another difference occurred when I used the tape backup (see the sidebar).

The Micropolis drive features an internal 64-Kbyte buffer, an automatic park-and-lock mechanism, and a fully balanced rotary voice coil positioner that allows any orientation in mounting the drive. Because the drive has a mean time before failure of 150,000 hours, I wouldn't expect any problems (this has been confirmed by the uneventful day-to-day operation of the drive). Thus, while I've read other reviewers harping about the problems they've had with SCSI drives, I have been exceedingly pleased by how easy it was to replace my old drive and the performance improvement the new drive offered.

Another interesting feature of the Always SCSI interface board is a rear 25-pin DB25 connector with the same pinout as found on the Apple Macintosh. This allows standard interface cables to connect external SCSI devices.

Current retail pricing for the 1684 PC Pak is \$1,935, with a street price of \$1,349. This is probably a little more than the average PC user wants to pay for a primary disk drive but is very reasonable for the drive on a server. In fact, as much as I enjoy using this

drive, it makes more sense to use it as a network device — a subject for later review. Either way, you get speed and capacity in a well-made drive and controller bundled into a fully integrated installation package. For my money, it's well worth the investment.

Readers can contact Micropolis Corp. at 2121 Nordhoff St., Chatsworth, CA 91311; (818) 709-3300; or circle Reader Service Number 26.

— Richard Eckhouse, University of Massachusetts-Boston

Proscan PS-3000. If you are seeking a relatively low-priced, high-quality, color flat-bed scanner, you should take a serious look at the Proscan PS-3000. This \$995 device from Our Business Machines is a well-designed small-footprint machine that can serve your needs for either color or grayscale image processing. The scanner comes bundled with the Windows-based Picture Publisher image-capture and editing software that easily rivals packages that cost hundreds of dollars.

Scanner installation took only a few minutes, including the 8-bit ISA adapter board. Installation software checked to see whether any IRQ conflicts existed with other installed devices. (The board can be set to IRQ 2-5.) After I resolved one of these conflicts by changing a setting on the other scanner adapter board that I was testing, the machine worked perfectly.

Picture Publisher software consists of two functional modules. The input module is used to scan an image that can be enhanced and edited by the second module. The input module provides scanner control options that include brightness and contrast, "posterize" (to reduce the number of gray ranges), threshold, "tonemap" (to alter selected parts of the gray scale),

positive/negative, and quartertones (to affect highlight, shadow, and mid-tone values). Images can be previewed and captured in 24-bit color, gray scale, binary, or dithered, all affecting the resultant file size and subsequent edit-processing and file I/O durations.

Although I was not enamored with the Picture Plus image-editing module, I think that my feelings were based on its nonstandard menuing scheme. Also, for some reason, the edit software would not let me display images that I was editing in the full resolution and color offered by my installed Windows graphics adapter. To make matters even worse, those images were displayed in CGA-like resolution. This is not what I would have hoped for when using this scanner together with my 20-inch, noninterlaced, 1,024 × 768 display. Unfortunately, I was not given enough time to really exercise the software to see if I could improve on this situation.

OBM seems to position this scanner as an inexpensive color-image-capture device for higher end desktop publishing or multimedia computing applications. The bundled software supports these environments but does not provide any OCR function. Although other OCR packages do support the PS-3000, that function is severely limited by the fact that the machine does not have a sheet feeder. Nevertheless, considering that other full-page, 24-bit color scanners cost about twice the price, if you are in the business of preparing and editing full-color images, this is a very cost-effective flat-bed scanner.

Readers can contact Our Business Machines at 12901 Ramona Blvd. Ste. #J, Irwindale, CA 91706; (800) 433-1435; fax (818) 960-1766; or circle Reader Service Number 27.

— Sorel Reisman, California State University, Fullerton

International Microcomputer Software Inc. I recently looked at two of this company's latest products. The mouse in its serial version is a white, beautiful, three-button affair with a switch enabling it to work as either a Microsoft or Mouse Systems device. It has an adjustable resolution range of 290 to 1,450 dpi and comes with its own driver software along with utilities for hard-disk management, menuing, and painting. It even has screen savers for CGA, MDA, and EGA monitors, but surprisingly lacks a screen saver for VGA! Also included are a mouse pad, a DB9 to 25-pin

adapter, free product support, and a lifetime warranty.

The mouse driver has several good features, including uninstall. All are accessible through different command-line options. The driver is accompanied by a good program to allow testing of cursor position, buttons in either mode, and resolution.

The menuing program contains useful commands for hard-disk management. It allows access to up to 300 different programs. It also lets the user perform 30 file-management operations including list, view, move, copy, delete, and undelete without having to type the DOS commands. For easy editing, users can select the built-in editor or specify the editor of choice. Some data security is provided through user login and password protection.

BackupDirect is an optional add-on for hard-disk backup. Another IMSI-supplied utility, MenuDirect, does not support the mouse very well. Rather, the documentation tends to encourage keyboard operations.

The Image 72 paint program is provided by ATech. It provides tools for drawing freehand, lines, circles, rectangles, cutting and pasting, and grabbing a screen image. It can retrieve and store images directly in the formats supported by Dr. Halo, Halo DPE, Spellbinder, CCITT fax format, GEM, Ventura Publisher, MS-Windows Paint, PC Pagemaker, Harvard Professional, and several other programs.

A second IMSI product called VirusCure Plus helps protect the software. As everyone knows, computer viruses have become a nightmare for computer system administrators. These viruses range widely in their nature of attack on the computer and the type of harm they do. Several programs are available to fight these viruses. The power of these programs lies in their knowledge of existing viruses, capability to detect newer kinds of viruses, and the actions they can take for prevention and cure.

McAfee Associates is a big name in virus detection and removal. VirusCure is the result of combined efforts of McAfee Associates and IMSI. It can identify, prevent, and remove viruses from the computer memory, hard disks, and floppies. It can also restore the infected files to their original working condition, in most cases.

The software, as distributed, has two major components: Cure and Protect. The Cure is a virus detection and removal program. When it detects a virus, it prompts users with the name

of the virus and allows them to remove it or to terminate the scan. If the removal option is chosen, Cure removes the virus and attempts to repair the infected files. It claims to repair almost all .COM files and about 80 percent of the .EXE files. All boot sector and partition tables can be repaired successfully. Cure can also run in the batch mode, in which user input is requested only when a virus is detected.

The Protect memory-resident program occupies 25 Kbytes of RAM and alerts users when a virus tries to penetrate the computer.

Presently, VirusCure Plus can recognize 230 known viruses. It can also detect unknown viruses, which is a very important feature, since some public domain and commercial programs "know" about 600 existing viruses. While scanning the disks, it searches through executable files and scans overlay files.

IMSI documentation could be improved significantly. I couldn't get the installation to work on either product and had to simply copy the files into a directory. The installation process for the menuing program was incorrect. Also, the package I received did not have BackupDirect, and Image 72 would not run on my machine. The manual provided no insight into what I might be doing wrong with the latter. And, while the hardware requirements have not been specified, supposedly the software runs on most DOS systems. The program was tested on an IBM PC AT-compatible with a 50-Mbyte hard disk, running under MS-DOS 5.0.

Although IMSI has a help line — (415) 454-7101 — it's easier to get help on the fax line (415) 454-8901 or on their BBS number (415) 454-2893. The help line is an answering machine that collected my calls for two weeks without returning one of them.

The IMSI sells for \$69.95. The VirusCure Plus is available in both 3.5- and 5.25-inch formats for \$99.95. Registered users get free updates for two years via BBS. For an additional \$30.00, the updates are mailed to the user for the same two-year period.

Readers may contact International Microcomputer Software Inc. at 1938 Fourth St., San Rafael, CA 94901; (415) 454-7101; fax (415) 454-8901; or circle Reader Service Number 28.

— Faisal Saeed, Oklahoma State University, Stillwater

The Wrist Pad. I have had the Wrist Pad by Silicon Sports on my desktop

for one month, and I have heard only one question: "Where can I get one?" Every person who has seen it wanted one, and so will any person who has ever worked overtime at a keyboard. The Wrist Pad is a 0.75 x 4.0 x 19.5-inch pad of nylon-coated neoprene, the material used in making such sports gear as wet suits and bicycle handlebar pads.

The design creates a resting surface that maintains the hands in an unstressed position, decreasing arm strain and minimizing compression of tendons and nerves in the arms and wrists.

In addition to the comfort it provides, the Wrist Pad might be helpful in preventing repetitive stress injuries. Current computer designs require users to sit with their arms in a tiring position and to hold their wrists in a bent-back position that can irritate the tendons, ligaments, and nerves. The Wrist Pad is designed to reduce stress to the hands, wrists, and arms. The manufacturer carefully notes that the Wrist Pad will not reduce injury all by itself but must be "part of an overall program of training, work organization, and equipment."

It is also important to position the desktop so that arms are parallel to the ground and wrists are straight. Chair, table, and keypad surfaces must be free of hard or pressing edges. I can't speak to the medical effects of the Wrist Pad. All I know is that after using mine for a month, I am not going to give it back. In fact, I am going to buy a second one to use on my home computer. Wrist Pad is available for \$19.95.

Readers can contact Silicon Sports at 1180 Sherman Ave., Menlo Park, Calif. 94025; (415) 854-1456 or (800) 243-2972; or circle Reader Service Number 29.

— Betsy Schwartz, University of Massachusetts-Boston



The wristpad.

NEW PRODUCTS

Contact or send releases to Christine Miller, Computer, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1264; Compmail, s.c. miller.

Multimedia from Big Blue

IBM has announced two multimedia kits for the PS/2. The training and marketing kits come with either the 8516 Touch Display or a 3.5-in. rewritable optical drive. Optional features include speakers, headphones, videodisc players, and instructional documentation.

The kit for such internal training applications as management, sales, and technical skills includes the M-Motion Video Adapter/A, the M-Control Program, and the 8516.

The marketing kit for desktop sales and merchandising uses Operating System/2 and includes an optical drive, a SCSI adapter, M-Audio Capture and Playback Adapter/A, and the runtime version of the Audio Visual Connection (AVC) 1.04.

The enhanced version of AVC has added support for the Extended Graphics Array graphical mode, while the M-Motion card provides window-within-a-window capabilities and multiple, concurrent motion windows.

IBM also plans to market the Media-source General Topic Library, a CD-ROM disk that contains 1,500 photographic images, 60 minutes of stereo-quality music.

The training kit is available for \$2,870 for a limited time. The marketing kit is available for \$2,110.

Reader Service 35

Computer tutor

The Xerox Reading Tutor from CSP Communications scans books or other typeset information, converts the material, and reads aloud to students. It incorporates a high-resolution scanner with Kurzweil's text-recognition software, a reading instruction format, and synthetic speech. Users can adjust reading speed to repeat a section, hear words pronounced, or spell words aloud.

Applications include adult literacy programs and classes for the visually impaired.

Reader Service 36

Video with Targa+

Truevision Inc. has announced VideoMaker+ software for generating video productions and desktop presentations. It uses Truevision Targa+ graphics for ISA-bus IBM PC ATs and PS/2s. A Windows 3.0 driver is included with each board.

Users can define a sequence of events containing either live video or computer-generated graphics. Transition capabilities include wipes, pushes, venetian blinds, and checkerboards. Effects include capture, videostrobe, solarization, motion blur, and feedback.

Images can be enhanced or re-

touched at the pixel level and output at broadcast quality to videotape, film recorders, or color printers.

VideoMaker+ requires a VGA monitor that can display RGB, S-Video, or Composite formats and work with NTSC and PAL standards. The system should also include a Microsoft mouse, a video-source camera or VCR, and DOS 3.2.

The software is included with Targa+ purchases and available to current users at a nominal fee. The Targa+ starts at \$1,795.

Reader Service Number 37

Make movies on your Mac

SuperMac Technology has announced a family of QuickTime-compatible digital video products for moviemaking on the Macintosh. QuickTime is Apple's media-integration system.

The products include VideoSpigot, a capture-and-playback system for the Mac LC and IIx; VideoSpigot Pro, with 8- and 24-bit graphics support and digital video on large-screen displays for the Mac IIx; and ReelTime movie-editing software.

VideoSpigot turns video into fully digital data that can be stored on an ordinary hard disk or optical cartridge. The Pro version provides color for publishing applications. ReelTime allows movie to be viewed on the Mac or exported into NTSC or PAL formats.

VideoSpigot costs \$699; the Pro version is \$1,899. ReelTime is bundled free for a limited time.

Reader Service 38

Media on the Iris

Silicon Graphics has developed a RISC PC for 3D graphics, DAT-quality audio, and video. The Iris Indigo system runs 1,500 packages and is compatible with the Advanced Computing Environment.

The Indigo computer incorporates a 33-MHz Mips R3000A processor, a 101-key keyboard, 8-bit graphics, 16-bit audio, Ethernet, SCSI, serial and parallel ports, and little- and big-endian byte ordering.

The system offers texture mapping, alpha blending, and the Iris Graphics Library application programmer's interface. It comes with a microphone

and real-time color video cards. Memory can be expanded to 1.3 Gbytes of disk space and up to 96 Mbytes of RAM.

Iris Indigo runs the Irix version of Unix and complies with Posix, X/Open, and FIPS 151-1 specifications. The X Window System is merged with the Iris interface and Adobe Display PostScript. A Motorola 56001 digital signal processor residing on the motherboard processes audio throughput.

Base price for the system is \$7,995, including 8 Mbytes of memory and a 16-inch color monitor.

Reader Service Number 39

Lights, camera, action!

Boxlight Corp. has introduced ShowMate Express for electronic slide shows using IBM PCs. This slide carousel on a disk lets users capture screens from — and combine them with — all standard PC programs. Menu-selection facilities let users capture and assemble screens in presentation order and incorporate special effects.

The resulting "show disk" can be used on other PCs or on Boxlight's presentation system.

The software costs \$49.95 and is free with the company's LCD projection panels for a limited time.

The company has also announced the 6.1-lb. Beacon notebook computer, which consists of a 386SX-20 CPU integrated with a VGA projection screen. It also comes with a math coprocessor socket, 4 Mbytes of RAM, a 40-Mbyte hard drive, and a 1.44-Mbyte 3.5-inch floppy drive. Users can detach the VGA screen and place it on top of a standard overhead projector.

The Beacon notebook costs \$3,495.

Reader Service 40

Wavefront software for the FX/2800

Visualizer Server Software from Wavefront Technologies is now available for the Alliant Computer Systems FX/2800 family of RISC-based parallel supercomputers. Users can create models interactively on standard workstations and send them to the supercomputer that serves UltraNet, FDDI, and Ethernet networks.

The software can be used with a "mental-ray" package for specialized ray-tracing features to create advanced realism in imagery. It also supports interfaces to standard CAD packages and such CAD formats as IGS, allowing users to import models directly from engineering systems.

The FX/2800 provides up to 28 i860 64-bit RISC processors, 4 Gbytes of memory, and multiple high-speed I/O channels.

Applications include computational chemistry, scientific visualization, television and movie production, and military simulation.

Reader Service 41

Sony meets desktop

Sony has developed a new Vbox video/computer interface that supports multimedia and desktop video software running on MS-DOS, Apple Macintosh, and Amiga computers. The interface, which automates the control of Sony consumer video devices, supports such packages as Director and MediaMaker from MacroMind, ShowMaker from Gold Disk, and AudioVisual Manager from TSE and Assoc.

Sony's Video System Control Architecture synchronizes the complex cueing of multiple video devices from within multimedia applications. The Vbox also automates the print-to-video process performed by the RGB/NTSC video-signal converter, which heightens portability.

The interface costs \$299.95.

Reader Service 42

LAN information exchange

Concentric Technologies has announced a family of Electronic Messaging and Information Exchange software in conjunction with LinkAge 3.1. LinkAge incorporates e-mail, group messaging, data conferencing, fax, and virtual terminal access tools and runs over LANs, WANs, and dial-up modems. The Unix-based server software provides these services to DOS users.

LinkAge can autonomously route CCITT Group-II faxes directly to user mailboxes without specialized hardware or continuous connection with the public telephone network.

System requirements for clients include IBM PC XT 386s with 512 Kbytes of free RAM, extended or expanded memory, and video adapters.

The server can be a Compaq 386 (or compatible or larger system) running Unix System V, R3.2, and have 4 Mbytes of RAM and 60-Mbyte storage.

Communications hardware varies with network selection.

Reader Service 43

Home, James!

Bright Star Technology has released At Your Service, which installs a talking actor, Phil, into Macintosh systems. Phil reminds users of appointments and special events, displays e-mail, and provides system information in English. He also monitors key-

Software upgrades its act

Performance Software has added advanced sound, graphic, and peripheral support features to Version 2.4 of its ACT III multimedia authoring system. The ACT III system allows non-programmers to use IBM-compatible PCs as a platform to develop and deliver disk-based communications. Users can create training tutorials and educational courseware, sales and marketing presentations, and demonstration disks.

The program includes PC Paintbrush and contains its own animation facility for text and graphics. Authors can incorporate extensive branching, high levels of user interaction, record-keeping, customized feedback, and personalization.

Version 2.4 adds support for a dozen third-part sound digitizing boards. Voyetra drivers allow presentations that contain recorded sounds and speech, play standard MIDI files, or play CD-ROM audio.

Support has also been added for Super VGA graphics modes, mouse input for both text and graphics screens, and automatic encryption of files.

Version 2.4 costs \$495; upgrades are available for \$125.

Reader Service 44

Portable computer talks

Animated Voice Corp. has developed a laptop computer for language-skill improvement. The 10-pound LapTalk MS-DOS computer comes with AniVox digitized voice I/O circuitry specifically designed for voice-interactive applications.

Features include a VGA display, a 40-Mbyte hard disk, and up to 4 Mbytes of expandable RAM.

In some applications, users can see realtime graphical displays of speech parameters for matching their voice accent, emphasis, and tonal quality to a model.

LapTalk is \$2,395.

Reader Service 45

board and mouse activity and suggests when it's time to take a break.

The program is System 7.0 compatible, works in both color and black and white, and costs \$59.95.

Reader Service 46

IC Announcements

Company, Model, Function	Comments	R.S. No.
ACC Microelectronics ACC-2046 Controller chip	Integrates memory, cache, and peripheral controllers in a single 208-pin device for 486DX/SX and 386DX-based AT systems. Supports 1 to 4 memory banks of 32-bit DRAM, using 256-Kbyte, 1-Mbyte, or 4-Mbyte DRAMs; Intel 80387-compatible math and Weitek 4167/3167 numerical coprocessors; 8 or 16-bit wide BIOS ROMs in 128 or 64-Kbyte spaces; and 64 Mbytes of DMA address space. Cost (1,000s): \$100, 25 MHz; \$125, 33 MHz; \$140, 40 MHz.	120
Elec. Eng. Software PC-OPT+ Circuit optimizer	Complete PC-based circuit simulator and optimizer software provides graphical environment that uses single mouse picks to move from editing netlist to optimizing circuit and viewing results. Algorithms let designers optimize any circuit they can simulate. Supports flexible performance specifications, interactive selection of parameters for change, and confirmation simulation. Cost: \$1,695.	121
I&J Fisnar VPP803A Vacuum tool	Vacuum pencil for chip pick-up and placement is adjustable up to 16 inches of mercury. Control box measures 3-3/4 x 7-1/2 x 2-5/8 inches; input air, 20 to 100 psi; input air hose, 6 ft. Includes three tip sizes.	122
Linear Technology LTC486, LTC487 Quad RS485 drivers	Micropower devices draw a maximum of 200 μ A current, support data transmission up to 10 Mbaud, operate from single 5V supply, handle a -7V to +12 bus common mode range, and feature 28-ns driver propagation delays with 5-ns skew. Supplied in 16-pin-molded DIP and 16-lead wide-body SOIC packages. Pin-compatible replacement for other RS485 drivers. Cost (100s): \$3.70.	123
American Neuralogix NLX230 Fuzzy logic controller	Single-chip, 40-pin fuzzy logic inference engine with on-chip rule memory. Supports 8 inputs, 8 outputs, and a resident base of 64 fuzzy logic rules. Intended for embedded applications; operates with or without a host microprocessor. Application Development System includes NLX 230, associated circuitry on a PC-compatible board, controlling software, and documentation. Cost: under \$4, NLX230; \$395, ADS.	124
Philips Semiconductors Optocouplers SMDs	All dual-in-line optocouplers are now available with gull-wing surface-mount lead-outs. Package and lead-out spacings are exactly the same as existing optocouplers, giving the new surface-mounted devices the same voltage specifications.	125
Texas Instruments TLC43046 Analog interface circuit	Single-chip DSP peripheral integrates user-programmable analog-input band-pass/low-pass filter with nominal bandwidth of 300-7,600 Hz. Maximum sampling rate of 25 kHz. Interfaces directly with serial ports on TMS320 DSP devices; requires minimal glue logic for interface to serial ports of non-TI DSPs and serial-to-parallel circuitry for parallel ports. Three operating modes and selectable (sin X)/X correction for D/A conversion. Cost (1,000s): \$14.63.	126
Unitrode UC3823A/B, UC3825A/B Pulse width modulators	New generation of high-speed PWM circuits: "A" devices feature undervoltage lockout thresholds that turn on at 10V and off at 9V; "B" devices turn on at 16V and off at 10V. Start-up current is 100 μ A. Other features include trimmed oscillator discharge current and pulse-by-pulse current-limiting comparator. Commercial parts available in 16-pin DIL (suffix N) or SOIC (suffix DW) packages. Cost (1,000s): \$3.60, UC3823AN/BN; \$4.30, UC3825AN/BN.	127
Unitrode UC1054 Converter/regulator	Charge-pump IC generates and regulates positive and negative voltages between 3.5V and 15V for "spot" power. Provides up to 100-mA output with 1.1V typical voltage loss over the full range. Can be paralleled for higher currents. Features internal oscillator with nominal frequency of 25 kHz. Available in commercial (C) and military (M) forms. UC1054C is available in plastic (N) and ceramic (J) 8-pin DIPs. Cost (1,000s): \$2.10, UC1054CN.	128
VTC VM7301, VM7401 Channel chip set	VM7301 is a data separator, synthesizer, encoder-decoder chip; VM7401 is a pulse and servo peak detector. Combined maximum power is less than 500 mW in the read/write modes and less than 10 mW in the sleep mode. The chip set supports a 24-Mbit/sec data-transfer rate and operates on a single +5V-only power supply. Both circuits are packaged in 44-pin quad flat-packs; VM7301 also comes in a 34-pin SOIC package. Cost for set (1,000s): \$23.45.	129
Xicor XM28C4096, XM28C080S EEPROM modules	The XM28C4096, in a 40-pin DIP module, comprises four X28C010 128K x 8 EEPROMs mounted on FR-4 substrate; it is configured 256K x 16 and features two separate chip-enable inputs to select between two 128K x 16 memory banks. The XM28C080S, a 36-pin SIP module, comprises eight EEPROMs and a high-speed decoder; it is configured 1M x 8 and offered in two speeds: 180 ns and 250 ns. Cost (100s): \$940, XM28C4096; \$1,680, XM28C080S.	130

Third IEEE Symposium on Parallel and Distributed Processing



Sponsored by: IEEE-Computer Society
ACM-SIGARCH
IEEE-Dallas Section



Dallas, Texas - December 2-5, 1991

Chair, Steering Committee:
Behrooz Shirazi, Univ of Texas-Arlington, USA

Steering Committee:
Said Bataineh, LSU, USA
Prasenjit Biswas, TI, USA
Eliezer Dekel, Univ of Texas-Dallas, USA
Ali R. Hurson, Penn State U, USA
Kishna Kavi, Univ of Texas-Arlington, USA
Ioannis Tollis, Univ of Texas-Dallas, USA

General Chair:
Sartaj Sahni, Univ of Florida, USA

Program Chairs:

Nakoto Amamiya, Kyushu Univ, Japan
Greg Pflister, IBM Corporation, USA
Vijaya Ramachandran, Univ of Texas-Austin, USA
Ian Watson, Univ of Manchester, UK

Program Committee:

Hosame Abu-Amara, Texas A&M, USA
Tadashi Ae, Hiroshima Univ, Japan
Tilak Agerwal, IBM Corp, USA
Laxmi Bhuyan, Texas A&M, USA
Bill Carroll, Univ of Texas-Arlington, USA
Henry Dragon, Univ of Texas-Austin, USA
John Fao, Lawrence Livermore, USA
Joydeep Ghosh, Univ of Texas-Austin, USA
Susanne Hambrusch, Purdue Univ, USA
A(Tony).J.Hey, Southampton Univ, England
Ray Juzz, RADC-AFOSR, USA
Michael Merritt, AT&T Bell Labs, USA
Les Miller, Iowa State Univ, USA
Rikio Onal, NTT, Japan
Simin Pakzad, Penn State Univ, USA
Ian Parberry, Univ of North Texas, USA
Behrooz Parhami, Univ of California-SB, USA
Girish Pathak, Xerox, USA
Greg Pflister, Univ of Texas-Austin
Donna Quammen, George Mason Univ, USA
Alejandro Schaffer, Rice Univ, USA
H.J. Siegel, Purdue Univ, USA
Barbara Simons, IBM Almaden, USA
Hal Sudborough, Univ of Texas-Dallas, USA
Kam-Fai Wong, ECRC GmbH, Germany
Yoshinori Yamaguchi, ETL, Japan
Yanjun Zhang, Southern Methodist Univ, USA

Finances: Randy Chow, CIS, Univ of Florida, USA
Publicity: Sarjaj Ranka, Syracuse, USA

Local Arrangements: Y. Paul Chiang, Texas Instruments, USA
Registration: Pearl Wang, George Mason Univ, USA

Publications: Sadeq Davari, Univ of Houston-Clear Lake, USA

This symposium provides a forum for the presentation and exchange of current work on a wide variety of topics in parallel and distributed processing including:

Computer Architecture	Programming Languages	Operating Systems
Neural Networks	Interconnection Networks	Database and Knowledge-base Systems
Artificial Intelligence	Parallel Algorithms	VLSI Systems Design
Simulation and Modeling	Distributed Computing	Scheduling

The technical program (December 3-5) features 94 long and 24 short papers, a plenary keynote speech, and a plenary panel discussion on "How can models promote mainstream parallelism?", moderator: H.J. Siegel, Purdue University.

Full-day Tutorials (December 2):

T1: Parallel Processing Algorithms & Systems, H.J. Siegel, Purdue University.
T2: Parallelism Management: Synchronization, Scheduling, & Load Balancing in Parallel & Distributed Systems, Behrooz Shirazi & Krishna Kavi, Univ of Texas at Arlington.

Hotel Reservations: Please place your reservations directly with Doubletree Hotel at Lincoln Center, 5410 LBJ Freeway, Dallas, TX 75240, Tel: (214) 934-8400, Or (800) 528-0444. You must mention the symposium (SPDP) in order to receive the special symposium rates (\$74/night - single room, \$84/night-double room.) Please check with the reservations desk for the applicability of other special rates, such as those available to AAA members. Reservations should be made before November 18, 1991. After this date, reservations are subject to availability.

Airline Reservations: As the symposium's official carrier, American Airlines offers a special meeting saver fare that allows a 40% discount off your round trip, unrestricted day coach fare, or 5% off any promotional fare. Please call 1-800-433-1790 for more information and for placing your reservations. Ask for star #: 02N1PG.

PLEASE SEND REGISTRATION FORM AND PAYMENT (payable to SPDP) TO: Dr. Pearl Wang, Symposium on Parallel and Distributed Processing, Computer Science Department, George Mason University, 4400 University Drive, Fairfax, VA 22030-4444, Tel: (703) 993-1530.

The symposium fee includes symposium proceedings, banquet, and social hour. Student registration does not include the symposium banquet. (Advance Registration: Before 11/20/91.)

	Symposium	Tutorial
ACM/IEEE Members:	Advance US\$240, On-site US\$290	Advance US\$190, On-site US\$240
Non-Members:	Advance US\$300, On-site US\$350	Advance US\$250, On-site US\$300
Full-time Students:	Advance US\$100, On-site US\$120	Advance US\$100, On-site US\$120

ACM Member: _____ IEEE Member: _____ Non-Member: _____ Student: _____

Symposium: \$ _____ IEEE/ACM No.: _____
Tutorial: \$ _____, Specify choice of tutorial: _____
Total: \$ _____ Student ID: _____

Check or bank draft enclosed (payable to SPDP), _____ Credit Card (VISA or Master Card ONLY)
VISA _____ or Master Card _____ No.: _____
Expiration Date: _____ Signature: _____

Last Name: _____ First Name: _____ Initial: _____

Organization: _____

Address: _____

City, State, Zip/Country: _____

Telephone: _____ E-mail: _____



THE INSTITUTE OF ELECTRICAL
AND ELECTRONICS ENGINEERS, INC.

Microsystem Announcements

Company, Model, Function

Comments

R.S. No.

- | | | |
|---|---|-----|
| ADAC
5508Super HR
Analog input module | Half-size PC XT/AT/386 and LapTop-compatible Direct Connect Series data-acquisition boards measure 3.9×6 inches and use an AD converter IC that ensures 16 bits of data for every conversion. Detachable screw terminations eliminate cabling and external screw terminal pads. Shipped with Direct View board tutorial and data-acquisition software. Cost: \$1,295. | 135 |
| ATI Technologies
Audio FX,
VGAudio FX
Multimedia boards | Eight-bit stereo DACs and ADCs, automatic adjust filters, and dynamic level controls for audio recording and playback with effective dynamic range up to 100dB. Compatible with Microsoft Multimedia Level I extensions to Windows. Internal multivoice synthesizers; MIDI interface to external devices optional on AudioFX, standard on VGAudio FX; graphics controller on VGAudioFX. Cost: under \$400, AudioFX; under \$600, VGAudioFX. | 136 |
| ATI Technologies
2400etc/i+SendFax
Internal fax modem | Complies with V.42-bis, MNP 5 data compression, and V.42 error-control standards to transmit and receive fax data at speeds up to 9,600 bps. Multiple interrupt support prevents serial port conflicts with other peripherals. Software interface lets users schedule transmission times and perform other functions. Compatible with other Class 1 software. Cost: \$199.95. | 137 |
| Bridgenorth
BN3000, BN3216
PC signal-processing
system | The BN3000 32-bit floating-point DSP processor board combines with the BN3216, an instrument-quality, two-channel, 16-bit analog interface module that connects directly to a parallel I/O port on the BN3000. Provide an integrated signal-processing system for acquisition, processing, and manipulation of large data sets for signal analysis, waveform generation, digital audio, and transfer function analysis applications on PCs. Cost: \$2,995, BN3000; \$995, BN3216. | 138 |
| Du Pont Pixel
PX10, PX20
Visual processing
boards | Single and multiple board configurations (PX10 and PX20) extend Power Fusion product line for integrated graphics, imaging, and numerics to desktop platforms with Sbus, MCA, AT/EISA, and VME architectures. PX20 uses PXbus to support parallel processing across multiple boards and displays. Includes software tools for applications development. Single and multiple slot configurations. | 139 |
| Hydra Systems
Andor One
PC-to-Macintosh
converter | Add-in board with TSR software allows PCs ranging from XT's through 486's to run Macintosh software. Users can switch between modes. PC's 3.5-inch floppy drive directly reads and writes Mac floppy disks. Includes an AppleTalk-compatible RS-422 connector and a licensed version of Word-for-Word/Mac utility for translating documents. Cost: \$995. | 140 |
| Nth Graphics
Nth TV
Video board | Display controller and video board for AT- and EISA-bus PCs generates true-color images and converts to NTSC-standard video output for animation. Provides 24-bit true-color drivers for AutoShade 2.0 and Autodesk 3D Studio, and 16-bit and 8-bit drivers for Windows 3.0. Includes high-resolution, antialiasing capability. Supports all VTRs. Cost: \$1,495. | 141 |
| Photon and Synesis
VideoGenesis/24
Frame buffer adapter | Uses a 16-bit Micro Channel bus and a single slot to provide 24-bit true-color video output in standard formats for IBM 6000s. Includes onboard RS-422/RS-232 serial I/O port for control of video peripherals and AIX device driver. Available in entry-level NTSC, broadcast NTSC, and PAL models. Cost: \$2,495, entry-level NTSC; \$3,595, broadcast-quality NTSC. | 142 |
| R.C. Electronics
RC-128
Waveform generator | AT plug-in card configured with 8 digital output ports and additional TTL ports lets users generate 8 separate waveforms at rates ranging from 1 point/hr to 250 KHz. Output in 1, 2, 4, or 8 channels. Waveform designs can range from 2 to 64K points, shared among the channels, with output in continuous or burst mode. Waveform resolution is 12 bits with a deglitch filter. Includes two software-programmable 4-pole filters for each channel. Cost: from \$2,895. | 143 |
| Specialix
RIO
I/O controller | Host cards for AT, Micro Channel, and EISA architectures based on Inmos T225 transputer support up to 128 ports at sustained throughput levels of 57.6 Kbaud full duplex. Four 10-Mbit/sec data channels link host to remote terminal adapters that contain own T225 transputer to power 4-Mbit/sec data channels on the host card. Cost: \$795, host cards; \$800, RTAs. | 144 |
| SuperMac
Spectrum/8×24 PQD
and PQD si
Color graphics cards | Single-slot boards for the Macintosh include a video controller that speeds graphics from the card to the screen at 40 Mpixels/sec and a custom-designed pixel accelerator that processes up to half a billion pixels/sec. Both cards support up to 8-bit color for 21-in. displays, 16-bit color for 16-in. displays, and 24-bit color for 13- and 12-in. displays; up to 8-ft.-sq. desktop resolution (4,096 × 1,536 pixels); and up to 200 percent hardware pan and zoom. Cost: \$1,399. | 145 |

Preliminary Program -- 12th IEEE Real-Time Systems Symposium

December 3-6, 1991

Sheraton-Gunter Hotel, San Antonio, TX 78205

Gen. Chair: Jane W.S. Liu - University of IL-Urbana/Champaign -- Program Chair: Lui Sha - SEI-CMU

Tuesday, December 3, 1991

Workshop on Architecture Support for Real-Time Systems

This workshop will provide a forum for discussion for researchers and practitioners on the architectural aspects of real-time systems. Authors should submit 4 copies of an extended summary of no more than 5 double-spaced pages by November 3, 1991 to one of the co-chairs Y.H. Lee, University of Florida, Gainesville, FL 32611 or C.M. Krishna, Univ of Massachusetts, Amherst, MA 01003. See Advance Registration below.

Wednesday, December 4, 1991

Software and Systems I -- Chair: Doug Locke

Towards an Automatic Synthesis System for Real-Time Software,
Tobiah E. Smith, Dorothy E. Settiff
RTC: Language Support for Real-Time Concurrency
Victor Wolfe, Susan Davidson, Insp Lee
Program Representation and Translation to Support Predictability of Real-Time Systems, Douglas Niehaus
A Partial Evaluator for the Maruti Hard Real-Time System,
Vivek Nirke, William Pugh

Short Presentations of Current System Developments,

Chair: Bill McDonald

Specification and Verification -- Chair: Insp Lee

Specification and Verification of Real-Time Embedded Systems Using the Time-Constrained Reactive Automata, Azer Bestavros
Clairvoyance, Capricious Timing Faults, Causality, and Real-Time Specifications, Douglas A. Stuart, Paul C. Clements
A Requirements Specification Method for Adaptive Real-Time Systems, Alice H. Muntz, Randall W. Lichota

Database -- Chair: Sang Son

On Using Priority Inheritance in Real-Time Databases, Jiandong Huang, J. A. Slankovic, Kithi Ramamritham, Don Towsley
Enhancing Real-Time DBMS Performance with Multiversion Data and Priority Based Disk Scheduling, Woosang Kim, Jaldeep Srivastava
Earliest Deadline Scheduling for Real-Time Database Systems, Jayant R. Haritsa Miron Livny, Michael J. Carey

Thursday, December 5, 1991

Real-Time Scheduling I -- Chair: Ragunathan Rajkumar

On The Competitiveness of On-Line Real-Time Scheduling, S. Baruah, G. Koren, B. Mishra, D. Mao, A. Raghunathan, L. Rosier, D. Shasha, F. Wang
Fixed Priority Scheduling of Periodic Tasks with Varying Execution Priority, Michael Gonzalez Harbour, Mark H. Klein, John P. Lehoczky
A Priority Ceiling Protocol for Multiple-Instance Resources, Min-H Chen, Kwai-Jay Lin

For hotel reservations call the Sheraton-Gunter at 800-222-4276 in San Antonio. Be sure to ask for the 12th RTSS rate of \$70 for a single and \$75 for a double. Room reservations must be made by November 22, 1991 to receive the special rate.

For further information on the Workshop or Symposium you can call Linda Buss at (612)659-2000 or fax your request to (612)653-2230. Registration should be mailed to Linda Buss, C/O Porous Media Corporation, 1350 Hammond Rd., St. Paul, MN 55110.

Workshop on Architecture Support for Real-Time Systems held on Dec. 3, 1991 will be advanced registration only.

**** ADVANCE REGISTRATION ONLY - by Nov. 15 ****

IEEE Member -- \$50.00 Non-Member -- \$65.00 Full-Time Student -- \$20.00



Short Presentation of Current System Developments,

Chair: David Pruett

Benchmarks, Standards and Case Studies --

Chair: Hide Tokuda

Real-Time Features for Ada 9X, Ted Baker, Offer Pazy
Building a Predictable Avionics Platform in Ada: A Case Study, C. Doug Locke, David, R. Vogel, Thomas J. Mesler
Early Experience with POSIX 1003.4 and POSIX 1003.4A
Bill O. Gallmeister
Harstone Distributed Benchmark: Requirements and Definitions,
Nick Kamenoff, Nelson Weideman

Fault Tolerance -- Chair: Jay Strosnider

The Transient Server Approach to Scheduling Time-Critical Recovery Operations and Periodic Workloads on a Shared Resource,
Sandra Ramos-Thuel, Jay K. Strosnider
Using Objects and Actions to Provide Fault Tolerance in Distributed, Real-Time Applications, S.K. Shrivastava, A. Waterworth

Friday, December 6, 1991

Software and Systems II -- Chair: Kevin Jeffay

Run-Time Monitoring of Real-Time Systems, Sarah E. Chodrow, Farnam Jahanian, Marc Donner
An Object Oriented Model for Parallel and Reactive System, A. Attoui, M. Schneider
MACS: A Predictable Architecture for Real-Time System, Bryce Cogswell, Zary Segall
Load Sharing with Consideration of Future Task Arrivals in Heterogeneous Distributed Real-Time Systems, Chao-Ju Hou, Kang G. Shin

Short Presentations of Current System Developments

Chair: Walt Heimerdinger

Network and Distributed Systems -- Chair: Jack

Stankovic

The Diffusion Model Based Task Remapping for Distributed Real-Time Systems, Morikazu Takegaki, Hiroo Kananaru, Masaya Fujita
Version Selection Schemes for Hard Real-Time Communications, Nicholas Malcolm
Efficient Synchronization of Clocks in a Distributed System, Sampath Rangarajan, Satish K. Tripathi

Real-Time Scheduling II -- Chair: K.J. Lin

On Non-Preemptive Scheduling of Periodic and Sporadic Tasks, Kevin Jeffay, Donald F. Stanat, Charles U. Martel
Optimization in Real-Time, Lon-Chan Chu, Benjamin W. Wah
Load Adjustment in Adaptive Real-Time Systems, Tei-Wuei Kuo, Aloysius K. Mok

Registration Form: 12th RTSS

Advance (Before November 15, 1991) Late (After November 15, 1991)
IEEE Member \$210.00 IEEE Member \$260.00
Non-Member \$290.00 Non-Member \$340.00
Full-Time Student \$ 75.00 Full-Time Student \$ 90.00

Name: _____
Affiliation: _____
Address: _____
City: _____ State: _____ Zip: _____
Country: _____
Phone Number: _____ E-mail: _____
IEEE Member Number: _____ Amount Due: _____
Checks should be made payable to the IEEE Computer Society - 12th RTSS.

Cancellation Policy: Registration fee will be refunded (less a \$50 processing fee) only if written cancellation is received by November 22, 1991.

Input requested for national database on software reliability

A panel initiated by the American Institute for Aeronautics and Astronautics is examining software reliability engineering estimation and prediction and is seeking participation by companies working on large software development projects of all types.

The effort involves developing and maintaining a national repository of software project data for use in system-planning activities and research projects. The database will be housed at the Data and Analysis Center for Software.

In addition, the panel and a working group are writing a handbook that includes recommended models, tools, and data collection methods.

Data sets collected will be used to tune the various models supporting software reliability estimation and prediction. Contributors will have access to the repository to support their own software reliability efforts.

The AIAA Working Group on Software Reliability, chaired by David Siefert of NCR Corp., requests that participants provide the following information on each project:

Project data.

- name of each life-cycle phase (e.g., requirements definition, design, code, test, operations);
- start and end date for each life-cycle phase;
- effort expended (in staff months) during each life-cycle phase;
- remarks about the development schedule (e.g., replans, problems, corrective actions);
- average development team experience (in years);
- the most important requirements, design, code, test, and configuration management tools and/or methods used;
- number of organizations developing software for the project;
- the Cocomo development environment (i.e., organic, semidetached, or embedded);
- the Software Engineering Institute index of the development environment and the assessment method;
- the most important tool and model used for software reliability estimation.

Component data. For each system component (e.g., subsystem, element,

or module) provide the following:

- software size in terms of executable source lines of code as well as the number of comments and the total number of object instructions;
- the source language used;
- the name and model of the development and target hardware; and
- average and peak computer resource utilization (e.g., CPU, memory, and I/O channel utilization).

Dynamic failure data. For each failure provide the following:

- activity being performed when the problem was detected (e.g., testing, operations, maintenance);
- date and time of the failure;
- number of CPU hours since the last failure;
- number of runs or test cases executed since the last failure;
- number of wall clock hours since the last failure;
- number of test hours per test interval and the number of failures detected in the interval;
- test labor hours since last failure;
- severity of the failure (e.g., critical, major, or minor);
- type of failure (e.g., interface, syntax);
- method of fault/failure detection (e.g., inspection, system abort, invalid output); and
- unit complexity (e.g., McCabe Cyclomatic, Halstead SW Science) and size where the fault was detected.

Fault correction data. For each failure corrected with a software fix, provide the following:

- date and time the fix is available;
- CPU hours required for the fix;
- number of runs required to make the fix;
- wall clock hours used to make the correction; and
- labor hours required for correction.

Additional information. Participants should provide a contact who knows the project and can answer questions concerning how the data was obtained and how the project-specific terminology translates to the database terminology. The Data and Analysis Center for Software will provide continuous support to clarify the foregoing requirements and to aid in data collection. It

is not necessary to provide all the data listed above, but a subset is required.

To participate in this effort, contact Stephen Kelly, Data and Analysis Center for Software, Kaman Sciences Corp., 258 Genesee St., Suite 101, Utica, NY 13502-4627, phone (315) 734-3662, fax (315) 734-3699, e-mail steve@kaman.com.

Graduate fellowship applications invited

The Charles Babbage Institute is accepting applications for the Adelle and Erwin Tomash Graduate Fellowship, to be awarded for the 1992-93 academic year to a graduate student whose dissertation will address some aspect of the history of computers and information processing. Topics may be chosen from the technical history of hardware or software, economic or business aspects of the information processing industry, or other topics in the social, institutional, or legal history of computing. Theses that consider technical issues in their socioeconomic context are especially encouraged.

The fellowship may be held at the home academic institution, the Babbage Institute, or any other location with appropriate research facilities. The stipend will be \$10,000 plus an amount up to \$2,000 for tuition, fees, travel to the Babbage Institute and relevant archives, and other approved research expenses.

Applicants should send biographical data and a research plan containing a statement and justification of the research problem, a discussion of the procedure for research and writing, information on availability of research materials, and evidence of faculty support. Applicants should have three letters of reference, certified transcripts of college credits, and GRE scores sent directly to the institute. There is no special application form.

Send complete application materials for receipt by January 15 to the Charles Babbage Institute, University of Minnesota, 103 Walter Library, 117 Pleasant St. S.E., Minneapolis, MN 55455, phone (612) 624-5050.

Software firms focus on flexibility, innovation

Bob Carlson, Staff Editor

Three recent announcements from major firms are indicative of the scramble among software developers to meet the expectations of users being enticed by faster, cheaper, and more versatile computer hardware.

Sun Microsystems, through its SunSoft division, plans to make its operating system available for high-performance personal computers, thus going head to head with Microsoft, as well as IBM and Apple, which are collaborating on a new operating system. The Sun product, Solaris, is a version of AT&T's Unix operating system. Several PC manufacturers said they will support Solaris in addition to the more familiar operating systems.

Industry leader Microsoft, meanwhile, announced the creation of a basic research center that will look two to five years into the future to determine how people will be using computers. The advent of credit-card-sized computers and the increased sharing of information among digitally based machines appear to be two of the forces driving the effort. Microsoft views the center, which will employ up to 100 people, as a long-term commitment involving millions of dollars.

Sony, primarily associated with consumer hardware, is finding software development an increasing concern as it pursues the field of interactive multimedia. Recently formed Sony Electronic Publishing must provide software for the Play Station, a new video game machine; for Data Discman, a portable device for accessing information, due out by Christmas; and for CD-ROMs. Sony has also indicated that it will support the CD-I (Compact Disc-Interactive) machine, which will need a wide variety of entertainment and educational software if consumers are to spend the \$1,000-plus anticipated purchase price. It remains to be seen whether software development, purchases through third parties, or corporate acquisitions will dominate the firm's strategy.

With multimedia standards still under development, a blurring of the traditional lines between technologies, and an ongoing need for rigorous self-examination to define their own scope and methodologies, software companies can be expected to announce important changes with increasing frequency. Faced with uncertainty, they must be able to regroup, refocus, and change directions smoothly and quickly.

Computer Society NEWS

Editor: Guylane M. Pollock, Sandia National Laboratories, Division 1424, PO Box 5800, Albuquerque, NM 87185; phone, (505) 846-0040; Internet, gmpollock@cs.sandia.gov

Barry Johnson receives Terman Award

The IEEE Computer Society's second vice president, Barry W. Johnson, has been picked to receive the 23rd annual Frederick Emmons Terman Award, a memorial award sponsored by Hewlett-Packard and awarded by the American Society for Engineering Education. Terman, long affiliated with Stanford University, was the mentor of HP cofounders William R. Hewlett and David Packard.

The award is given each year to one individual under the age of 42 for significant contributions to the education of electrical engineers. Johnson, an associate professor of electrical engi-

neering at the University of Virginia, was selected from among several hundred nominees.

The award citation recognized Johnson's contributions to the technical community, along with his teaching, research, and text book, *The Design and Analysis of Fault-Tolerant Digital Systems*, which has been accepted as a course text at more than 65 universities.

In addition to an expense-paid trip to New Orleans for the presentation at the ASEE awards dinner, Johnson receives a gold medal, a bronze plaque, and a cash award of \$4,000.

Parallel and distributed information systems conference to debut

A new conference intended to bring together a diverse technical community and provide a single, cohesive, high-quality forum is set for December 4-6 at the Fontainebleau Hilton Hotel, Miami Beach, Florida. The First International Conference on Parallel and Distributed Information Systems will cover design, development, and utilization of information systems in a distributed environment, together with all aspects of applying parallel processing to information management.

According to General Chair Amit Sheth, the conference, subtitled "Issues, Architectures, and Algorithms," derives its strength from merging three meetings: the International Workshop on Database Machines; the International Conference on Databases, Parallel Architectures, and Their Applications; and the International Symposium on Distributed and Parallel Database Systems.

The conference is cosponsored by the IEEE Computer Society, ACM SIGMOD and SIGArch, and Florida International University. For additional program and registration information, contact the Computer Society Conferences Department, 1730 Massachusetts Ave. N.W., Washington, DC 20036.

Computer Society cited for use of marketing tools

A study published by the American Society of Association Executives Foundation has cited the IEEE Computer Society as one of 20 successful associations that use marketing tools to guide current and future programs.

The study, *Market-Driven Management: Lessons Learned from 20 Successful Associations*, conducted by Donald Norris, lists such market-based tools as member surveys, focus groups, surveys of publication subscribers and authors, meetings with members to evaluate services, and member hotlines, noting that the Computer Society has used many of these tools. For example, the society used results from a 1987 telephone survey of current, former, and potential members to revise many of its operations. That survey was repeated this year.

Norris notes that the Computer Society became more market driven after experiencing a decline in membership in the mid-1980s but no decline in demand for services and programs. "The Computer Society could not undertake all its potential programmatic opportunities and had to start making programmatic choices, which in turn sharpened concern about what membership and other markets wanted and needed," Norris said.

STANDARDS

System Performance Evaluation Cooperative benchmark suite available

The System Performance Evaluation Cooperative's second benchmark suite, designed to provide users with a thorough and meaningful method of evaluating the performance of high-end computers and workstations, is now available. The new dual-benchmark suite, SPEC SDM Release 1.0, can be used with a wide range of architectures and environments, according to SPEC, a nonprofit consortium representing 22 computer companies.

The two benchmarks in the System Development Multitasking release are Software Development Throughput (SDET), for measuring Unix-based systems, and Kenbus1, for measuring Unix computers used in research and development. Both benchmarks exercise a variety of system resources, such as CPU, memory, disk I/O, and operating system services.

"These benchmarks give both vendors and users a fair yardstick to measure a computer system's performance, running in a real software development environment," said Kaivalya Dixit, SPEC president. "They are well suited for the multitasking/multiprogramming environment found at many customer sites."

Release 1.0 measures performance by gradually increasing the work load on the system until a constraint is reached in some system component. Running SDM Release 1.0 is an iterative process, since the bottlenecks identified can be removed by adding system resources.

The gradual increase in work load is achieved by increasing the number of concurrent copies of scripts. Each copy of the work load is called a script, which is synonymous to a Unix user process. A script stresses various system components. Each SDET script executes about 150 Unix commands, including spell, nroff, diff, make, and find. Each Kenbus1 script executes 18 Unix commands, including cc, cat, grep, mkdir, and rm.

The throughput metric, defined as the total amount of work done in a given time, is called scripts/hour. A throughput curve is plotted as scripts/hour against the number of scripts. The highest number of scripts/hour measured, called the peak value, is reported as the peak system throughput.

To reduce the variability of results for a given system, the benchmarks are run at least three times, and the

arithmetic mean of the results is reported for both the peak scripts/hours and the full throughput curve. Analysis of the throughput curve profile provides information on peak performance, capacity, and multitasking capabilities and reveals system strengths and weaknesses, according to SPEC.

Several SPEC "benchathons" were held to test the SDM benchmark suite on a variety of computers, ranging from workstations to high-performance superservers.

The new suite complements SPEC's first benchmark suite, SPEC Release 1.2, issued in 1989 and mainly used to compare the CPU speed of different systems while running a fixed set of scientific and engineering applications.

For more information, contact SPEC at 2722 Merrilee St., Suite 200, Fairfax, VA 22031, phone (703) 689-9600, ext. 318.

Planning study group to meet during ITC 91

The IEEE Computer Society is the interim leader of an industrywide multiorganizational study group that will meet October 29 during the International Test Conference in Nashville, Tenn.

The Software Engineering Standards Long-Range Planning Study Group was formed during the Computer Society's 1991 Software Engi-

neering Standards Applications Workshop in San Diego, Calif., to promote implementation of effective standards that fit together in content and purpose, according to Leonard L. Tripp of Boeing Commercial Airplane.

To carry out its mission, the study group will

- (1) identify long-range development and harmonization requirements for software engineering standards and
- (2) prepare and coordinate a plan for meeting the requirements.

A set of guidelines is being prepared to focus preparation of the requirements. Interested parties are encouraged to review and comment on the guidelines.

If you would like to assist in guidelines preparation, make a presentation at the October 29 meeting, or attend the meeting, notify Leonard Tripp at Boeing Commercial Airplane, PO Box 3707, MS 6Y-07, Seattle, WA 98124, phone (206) 477-3326, fax (206) 477-3330, Internet ltripp@kyl1.profs.boeing.ca.com.

Moving?

PLEASE NOTIFY
US 4 WEEKS IN
ADVANCE

MAIL TO:
IEEE Service Center
445 Hoes Lane
Piscataway, NJ 08854

ATTACH
LABEL
HERE

Name (Please Print)

New Address

City

State/Country

Zip

- This notice of address change will apply to all IEEE publications to which you subscribe.
- List new address above.
- If you have a question about your subscription, place label here and clip this form to your letter.

CAREER OPPORTUNITIES

RATES: \$12.00 per line, (ten lines minimum). Average five typeset words per line, eight lines per column inch. Add \$10 for box number. Send copy at least one month prior to publication date to: **Marian B. Tibayan, Classified Advertising, COMPUTER Magazine, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1264; (714) 821-8380; fax: (714) 821-4010.**

In order to conform to the Age Discrimination in Employment Act and to discourage age discrimination, **COMPUTER** may reject any advertisement containing any of these phrases or similar ones: "...recent college grads...", "...1-4 years maximum experience...", "...up to 5 years experience," or "...10 years maximum experience." **COMPUTER** reserves the right to append to any advertisement without specific notice to the advertiser. Experience ranges are suggested minimum requirements, not maximums. **COMPUTER** assumes that since advertisers have been notified of this policy in advance, they agree that any experience requirements, whether stated as ranges or otherwise, will be construed by the reader as minimum requirements only.

CLARKSON UNIVERSITY Electrical and Computer Engineering

Applications are invited for a tenure track faculty position as Assistant/Associate/Full Professor in the area of computer engineering. Candidates with backgrounds in distributed and parallel computation, artificial intelligence, image and signal processing, neural networks, and robotics are particularly encouraged to apply. Responsibilities include undergraduate and graduate teaching and development of a research program. A doctorate is required. Review of applications will begin on October 15th and will continue until the position is filled.

The department, consisting of 28 faculty members, offers programs at the B.S., M.S., and Ph.D. levels. Last year 122 EE bachelors, 20 CE bachelors, 15 masters, and 9 doctorates were awarded, and research funding reached more than one million dollars.

The department has research laboratories in artificial intelligence with 4 TI and 2 Symbolics LISP computers, neural computing with a 16 transputer system, VLSI design with 2 CALMA CAD systems, and robotics with a variety of industrial robots and vision systems. The department computing sub-network includes numerous SUN workstations. Other department laboratories include communications, lasers and optics, solid

state device fabrication, high voltage engineering, and dielectric breakdown.

Clarkson is an independent university specializing in engineering, science and management with an enrollment of 3300 students, including 400 graduate students. Located in northern New York, Clarkson is close to Lake Placid and the Adirondack Mountains.

Send applications to Professor Henry Domingos, Chairman, Department of Electrical and Computer Engineering, Clarkson University, Potsdam, New York 13699-5720. Clarkson is an Equal Opportunity/Affirmative Action Employer. Position No. 245.

D.E. SHAW & CO. Algorithmic Trading

D.E. Shaw & Co. is a small (several dozen employees), highly capitalized (over \$300 million in partners' equity), extremely successful Wall Street firm specializing in quantitative finance and computational trading. Computer scientists and system designers form the professional core of the firm, and not merely its support staff, and participate in its profits. Our technical staff includes both Ph.D.-level researchers recruited from Stanford, MIT, and other leading computer science departments and extraordinarily talented B.S. and M.S.-level system designers and "superhackers." It is our practice to compensate unusually gifted individuals at a level exceeding that of the market. Applicants should send resumes to Esther Treitel, D.E. Shaw & Co., 251 Park Avenue South, 15th Floor, New York, NY 10010.

AUBURN UNIVERSITY Earle C. Williams Eminent Scholar Chair in Electrical Engineering

Nominations and applications are invited for the Earle C. Williams Eminent Scholar Chair in Electrical Engineering. Candidates for this chair should have achieved national and international prominence in digital systems and/or microelectronics.

Applicants or nominees must have an earned doctorate, senior academic experience, and a documented record of distinction in university teaching and research. The successful candidate will be expected to provide intellectual leadership in his/her area of expertise for the Department of Electrical Engineering as well as enrich the scholarly environment at Auburn University.

Auburn University is located in the city of Auburn in east-central Alabama. This land-grant university enrolls more than 21,000 students, the largest on-campus enrollment in the state. The Department of Electrical Engineering, one of eight departments

within the College of Engineering, offers Bachelor, Master, Master of Science and Ph.D. degrees in Electrical Engineering. The department has a current enrollment of 939 undergraduate students and 100 graduate students. The 28 full-time faculty have an annual research expenditure of approximately \$2 million.

The Search Committee will begin its review of applications immediately. Interested candidates should submit: (1) a detailed resume, (2) a letter indicating an interest in the chair, the candidate's academic philosophy, and a brief statement of accomplishments in teaching and research, and (3) names and addresses of five references. Nominations should be submitted with the complete name, mailing address and telephone number of the individual nominated.

Applications and nominations should be sent to Professor J. David Irwin, Department of Electrical Engineering, Auburn University, AL 36849-5201. Auburn University is an affirmative action/equal opportunity employer. Applications from minority and female candidates are encouraged.

COMPUTER & INFORMATION SCIENCE

The Department of Computer and Information Science at Cleveland State University has a tenure track position in the areas of Computer Science or Information Systems. Responsibilities include teaching (two courses/quarter) & research. Salary is very competitive. **QUALIFICATIONS:** Ph.D. in Computer Science, Information Systems, or a closely related field. Close relationships to business, engineering, and other departments provide an environment conducive to research and consulting. The department offers undergraduate and masters degrees in Computer Science, cooperates with electrical engineering to offer the computer engineering option, and offers degrees in Information Systems at the undergraduate and DBA level. The university computing facilities include a fiber backbone network linking IBM mainframes, a VAX cluster, and college and departmental local area networks for both students and faculty. The university network is linked to Internet through the Ohio Research Network. Departmental and college local area networks include both personal computers & workstations. All faculty offices are equipped with personal computers linked to the networks. State of the art software is available on all computers. Inquiries and vita should be sent to: Chairperson, Computer & Information Science Dept., Cleveland State University, E 24th & Euclid Ave., Cleveland, OH 44115. Application Deadline: Until filled. Starting Date: September, 1992. Equal Opportunity Employer, m/f/h.

UNIVERSITY OF NOTRE DAME Faculty Positions

The Department of Computer Science and Engineering at the University of Notre Dame invites applications for tenure track faculty positions at all ranks. Applicants should have a doctorate in Computer Science, Computer Engineering, Electrical Engineering, or a related field. Candidates in all research areas are invited to apply. However, areas of particular interest within the Department are Parallel and Distributed Computing, Parallel Architectures, and VLSI. Applicants should have abilities and interests in teaching at the undergraduate and graduate levels, advising students, and conducting research. Rank and salary are negotiable. Interested persons should forward a complete resume, together with the names, addresses, and telephone numbers of at least three references, to:

Dr. Steven C. Bass, Chairman
Department of Computer Science
and Engineering
University of Notre Dame
Notre Dame, IN 46556
The University of Notre Dame is an Affirmative Action/Equal Opportunity Employer

THE OHIO STATE UNIVERSITY Department of Computer and Information Science

We are seeking highly qualified candidates for faculty positions at all levels and in all areas of computer science. Applicants should submit a vita and the names of at least three references to Chair, Faculty Search Committee, Department of Computer and Information Science, Ohio State University, Columbus, Ohio 43210-1277. The search committee will begin to consider applications December 1, 1991, and continue until the positions are filled.

The Ohio State University is an Equal Opportunity Affirmative Action Employer.

WASHINGTON UNIVERSITY

Washington University in St. Louis seeks qualified candidates for the position of Professor and Chair of the Department of Computer Science, with a desired starting date of July 1, 1992. We are interested in candidates with a strong research record, with a dedication to excellence in undergraduate and graduate education and with a demonstrated potential for administration and leadership.

The Department has an excellent undergraduate program as well as a strong and expanding graduate program. The primary research concentrations are in distributed systems, advanced communication networks and intelligent computer systems with an emphasis on visualization as a tool in each case. The Department plans to continue

building on these areas of strength as well as expanding into new areas. There are 15 regular faculty in the Department and 85 graduate students, as well as an excellent technical support staff and a large pool of affiliate faculty. Departmental laboratory facilities are very good and include a visualization laboratory, a systems prototyping lab, a NCUBE parallel computer, a variety of computer servers and ubiquitous workstations.

Washington University has a longstanding commitment to the principle that all candidates should be afforded equal opportunity regardless of age, race, sex or physical disability. Candidates must send a curriculum vitae and a list of references to: Professor C. I. Byrnes, Search Committee for the Computer Science Chair, Campus Box 1040, Washington University, One Brookings Drive, St. Louis, MO 63130

THE AMERICAN UNIVERSITY IN CAIRO Faculty Position

One Assistant, Associate, or Full Professor needed to teach, in English, upper division undergraduate courses in Computer Science. Ph.D. required. Rank, salary based on qualifications and experience. Two-year appointment (renewable) begins September 1992. For expatriates, housing, roundtrip air travel, plus schooling for two children included. Write with curriculum vitae to: Dean George H. Gibson, The American University in Cairo, 866 United Nations Plaza, Suite 5107, New York, New York 10017, preferably before November 15.

McGILL UNIVERSITY FACULTY OF MANAGEMENT Faculty Positions in Information Systems

The Faculty of Management has tenure track Assistant/Associate/Full Professor positions available in Information Systems commencing January 1992. We seek individuals with strong technical & managerial skills and having a distinguished research record or potential for the same with research/teaching interests in one or more of the following areas:

Modelling of Organizational and Inter-Organizational Information Systems, Systems Analysis and Design, EDI, Simulation.

The Faculty of Management has over 60 full-time faculty and offers B.Com, MBA and Ph.D. degrees. There are 1,400 Undergraduate, 400 MBA and 45 Ph.D. students. Salary is competitive.

Apply in writing to: Prof. Morty Yalovsky, Associate Dean - Academic, Faculty of Management, 1001 Sherbrooke St. West, Montreal, Quebec, H3A 1G5.

In accordance with Canadian Immigration requirements, priority shall be given to Canadian citizens and permanent residents of Canada.

DEPAUW UNIVERSITY Department of Mathematics and Computer Science Greencastle, IN 46135

Tenure track position teaching courses in undergraduate computer science major beginning August, 1992. Assistant or Associate Professor. Salary dependent on qualifications. Research is encouraged; commitment to excellence in teaching essential. Duties include teaching three courses each semester. Ph.D. in Computer Science or Mathematics required plus competence to teach several undergraduate courses in Computer Science.

DePauw is a selective liberal arts university with strong commitments to diversity and the sciences. Excellent fringe benefits and strong faculty development program.

Review of applicants will begin immediately; interviews will be scheduled during December, 1991, and will continue until the position is filled. Send vita, transcript, and three letters of recommendation to Gloria Townsend, chair of the search committee, at the above address. DePauw is an EO/AA employer. Women and minorities encouraged to apply.

SOFTWARE DEVELOPMENT ENGINEER

Design, document, implement and test software to implement network enhancements for the digital telecommunications switch customer. Debug and resolve software related problems reported by the customer. Design and code feature functionality on the digital switch. Test said features to ensure that all customer specifications have been met and that all code paths have been exercised. Interact with other engineers to communicate design strategies and feature interaction. Interact with management on technical and administrative issues. Make decisions with respect to software design, development, and problem resolution. Solve technical problems that affect the performance of the digital switch in regards to feature functionality in the area of feature development in Integrated Systems Digital Network (ISDN) and Common Channel Signalling System #7 (CCSS7) applications. Requires a Bachelor's degree in Computer Science and two years experience in job offered or two years related digital telecommunications software experience; or will consider a Master's degree in Computer Science with either three months Co-Op or project work experience in the design and development of software (systems) in lieu of the Bachelor and two years experience. Requires six credit hours in the following: software design in a high level language; network and protocols; and telephony. Must have three credit hours of CMS and Product Development Environment, 40 hour work week, \$47,000.00 per year. Apply at the Texas Employment Commission, Dallas, Texas or send resume to the Texas Employment Commission, TEC Building, Austin, Texas 78778, Job Order #6139127. Ad Paid by an Equal Opportunity Employer.

**UNIVERSITY OF CALIFORNIA
RIVERSIDE
Department of Computer Science**

Applications are invited for a position in Computer Science beginning July 1, 1992, or later. Ph.D. and excellence in research and teaching are required. Preference will be given to Systems (architecture, software). Exceptional candidates in other areas will also be given serious consideration. The position is open as to rank, but senior candidates (for Associate and Full Professor) are preferred.

The new faculty member will have an opportunity to participate in shaping a new and developing program in Computer Science.

The duties of the position include teaching at both the undergraduate and graduate levels, research, and participation in the Department and the Computer Science program.

Send all materials including curriculum vitae, publication list, and the names of at least three references to: Professor Marek Chrobak, Chairman, Computer Science Recruiting Committee, Department of Computer Science, University of California, Riverside, CA 92521. The pool of candidates will consist of all those whose files are completed by January 6, 1992. If the position is not filled from that pool of candidates, the pool will be expanded to include all those whose files are completed by February 24, 1992.

University of California, Riverside, is an Affirmative Action/Equal Opportunity Employer.

**UNIVERSITY OF NEBRASKA-
LINCOLN
Computer Science &
Engineering Faculty Positions**

Applications are invited for tenure-track positions in CSE or for joint appointments with Electrical Engineering or UNL Research Centers starting August, 1992. Rank and salary will be commensurate with experience. Hiring preference is at the Assistant Professor level. Women and minorities encouraged to apply.

Areas targeted for expansion are: (1) Database, (2) Operating Systems, (3) Programming Languages and Compilers, and (4) Software Engineering. Outstanding experimentalists in other areas will also be considered.

The department has 18 full-time faculty active in research and offers B.S., M.S., and Ph.D. programs. It has 60 graduate and 460 undergraduate students.

UNL has many computing resources linked by a comprehensive campus-wide network. It is the lead institution in the NSF-funded regional network, MIDnet and a node on the NSFnet backbone. UNL has state supported centers including the Center for Communication and Information Sciences and the Center for Technology Development and Decision Sciences.

Send resume, names of three references and letter describing current teaching and research interests by January 6, 1992 (or until suitable candidates apply) to:

Alvin J. Surkan, Search Chair
Department of Computer Science &
Engineering

UNIVERSITY OF NEBRASKA-LINCOLN
Lincoln, Nebraska 68588-0115

Affirmative Action/Equal Opportunity Employer.

**HUNTER COLLEGE OF CUNY
Chair,
Computer Science Department**

The Computer Science Department at Hunter College of The City University of New York is looking for a dynamic individual to be Professor and Chair of this growing, research-oriented department, to be effective September 1992. Candidates must have a Ph.D. in Computer Science or its equivalent, along with an exemplary record of scholarly activity, teaching, leadership and research in the field.

The department is relatively new, but has made great progress in its ten years. All of its full-time faculty are members of the Ph.D. Program in Computer Science at CUNY's Graduate Center, and Hunter has an in-house MA Program in Computer Science as well. Primary research areas are artificial intelligence, software engineering, graph theory, combinatorics and fuzzy logic. The possibility of collaborative research with nearby institutions is vast, as is also true of similar opportunities in the business world.

Hunter itself is centrally located on Manhattan's upper East Side and is convenient to all forms of public transportation.

Applicants should send a detailed C.V. and the names of three references to:

Dr. T.C. Wesselkamper, Acting Chair
Department of Computer Science
Hunter College of CUNY
695 Park Avenue
New York NY 10021

Hunter College is an Affirmative Action/Equal Opportunity Employer.

**McMASTER UNIVERSITY
Research Assistant**

Participation in research into cellular communication between intestinal smooth muscle cells. The research involves both mathematical modelling and physiological experimentation. Requirements are a Master's degree in engineering (with knowledge of electromagnetic theory and oscillator theory) and experience in physiological sciences (preferably electrophysiology of smooth muscle). Salary: \$28-30,000. For information contact: Dr. J.D. Huizinga, McMaster University, HSC-3N5C, 1200 Main St. W., Hamilton, Ont. Canada L8N 3Z5.

**RESEARCH SCIENTIST
IN SOFTWARE ENGINEERING**

This is a research position in Software Engineering in the Computer Science Department of the Naval Postgraduate School in Monterey, CA. Responsibilities include development and implementation of NSF-sponsored research software for the Computer-Aided Prototyping System (CAPS). The work involves design, Ada code generation, and scheduling algorithms for hard real-time and distributed software systems. Applicant must hold a Ph.D. degree in computer science and should have a good understanding of algorithms, compiler logic, and attribute grammars, as well as experience in Ada, C++, X-window applications and UNIX systems.

Send a detailed resume, an abstract of your recent research, and letters of recommendation to: Prof. Luigi, Department of Computer Science, Code CS/Lq, Naval Postgraduate School, Monterey, CA 93943. Telephone (408) 646-2735.

An Equal Opportunity/Affirmative Action Employer.

SOFTWARE SYSTEMS DESIGNER

Build parallel computer-based cardiovascular monitor and design and implement a relational DBMS for textual information. Experience with C, Unix, a Master's degree in computer science (or equivalent) and publications are required. Graduate courses in software engineering, parallel computing, biomedical signal processing, data structures, database design, and distributed database systems are required. Salary: \$32,600/yr. Send resumes to: Job Service Technical Unit, Connecticut Dept. of Labor, 200 Folly Brook Blvd., Wethersfield, CT 06109.

Refer to Job Order #3090538 (No calls).

**ACADEMIA SINICA
TAIWAN, REPUBLIC OF CHINA
Institute of Information Science**

Applications are invited for research position in Institute of Information Science, Academia Sinica, Ph.D. in Computer Science or closely related fields required. Demonstrable research ability necessary. Applicants for senior positions must have proven research record. All fields in Computer Science are welcome.

The Institute offers a good research environment. No duty of teaching. Facilities include a 32-node NCUBE, 2 parallel supercomputer, many SUN, SGI, and E&S workstations. An easily accessible ETA-10Q supercomputer is in the Academia Sinica.

Interested people please send application to Dr. Lin-Shan Lee, Director, Institute of Information Science, Academia Sinica, Taipei, Taiwan, 11529, Republic of China. Fax: (011-886-2) 782-4814.

UNIVERSITY OF HONG KONG
Reader/Senior Lecturer/Lecturer in
Computer Science and Information
Systems (Ref. 90/91-97)

Applications are invited for Reader-ship/Senior Lectureship/Lectureship posts in the Department of Computer Science. Candidates should have a higher degree in Computer Science, Information Systems or Computer Engineering, and strong interest in both teaching and research. Extensive experience in supervising M.Phil. and Ph.D. students and an excellent research record are essential. Well-qualified applicants are sought in all areas of Computer Science but especially in Information Systems, Logic Programming, Artificial Intelligence, Graphics, Computer Vision, Software Engineering, Database and Neural Network.

Annual salaries (under review) (superannuable) are on the scale: Reader HK\$453,780 - 602,880 (9 points) (approx. £35,860 - £47,650); Senior Lecturer HK\$435,000 - 584,340 (9 points) (approx. £34,380 - £46,180); Lecturer HK\$279,900 - 467,700 (11 points) (approx. £22,120 - £36,960) (Sterling equivalent as at July 1, 1991). Starting salary will depend on qualifications and experience. At current rates, salaries tax will not exceed 15% of gross income. Housing at a charge of 7.5% of salary, children's education allowances, leave, and medical benefits are provided.

Further particulars and application forms may be obtained from Appointments (39730), Association of Commonwealth Universities, 36 Gordon Square, London WC1H 0PF, UK; or from the Secretary, Faculty of Engineering, University of Hong Kong, Hong Kong (Fax (852)-5469142; E-mail: HUUR-ENG@HKUVM.BITNET).

Closes 22 November 1991.

MEMORIAL UNIVERSITY OF
NEWFOUNDLAND
Department of Computer Science
St. John's, Newfoundland, Canada
Faculty Position

Applications are invited for a tenure-track position at the rank of associate professor or professor. Candidates with established expertise in foundations of computer science, and in particular in the application of formal methods to computations and computational models, theory of programming languages, and parallel and distributed computations, are especially invited to submit applications.

Applicants should have demonstrated excellence in research and teaching in computer science. Duties and responsibilities include conducting research and supervision of graduate students as well as teaching graduate and undergraduate courses. Appointments will begin 1 January 1992 or at the earliest possible date thereafter. Salary will be commensurate with qualifications and experience. This position is subject to budgetary approval.

To be considered for this position please send your curriculum vitae and the names and addresses of three referees to: Search

Committee, Department of Computer Science, Memorial University of Newfoundland, St. John's, Newfoundland, Canada A1C 5S7; tel: (709) 737-8627, fax: (709) 737-2009; e-mail: pat@garfield.cs.mun.ca.

In accordance with Canadian Immigration regulations, consideration will first be given to those applicants who at the time of application are Canadian citizens or permanent residents of Canada.

INTERNATIONAL COMPUTER
SCIENCE INSTITUTE,
BERKELEY, CALIFORNIA
Postdoctoral and Visiting Positions

The International Computer Science Institute announces the availability of postdoctoral and visiting appointments for 1992 and beyond. The Institute is a non-profit basic research organization physically near and loosely affiliated with the University of California at Berkeley. Postdocs will work with the Institute and UCB faculty and staff on current projects. More senior visiting researchers can propose projects of any length and character. ICSI conducts research in several areas of parallel and distributed computation including theory, realization and applications of massive parallelism and the design of multimedia distributed systems and very high-speed networks.

Applicants should request application forms from:

Jerome A. Feldman, Director
International Computer Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704-1105 U.S.A.
Fax (415) 643-7684 (After Sept. 2, 1991
(510) 643-7684)

Internet/CSnet: info@icsi.berkeley.EDU

In order to save time, applicants should ask at least two references to write directly to the above address so that letters of evaluation reach ICSI at about the same time as the other application materials. Selection will be based on overall qualifications and compatibility with ICSI research plans. Applications from citizens of ICSI sponsor nations Germany, Italy and Switzerland are especially welcome.

ICSI is an Equal Opportunity Employer.

SYSTEMS ANALYST

Analysis procedures and problems to refine customers' statistical data and convert for electronic data processing. Confer with managerial personnel to clarify hardware & software in isolate problems; suggest changes and determine to extent to which automatic systems and coding technique can be used. Required M.S. in Applied Computer Science with 0 experience. Must be familiar with NOVELL ENVIRONMENT UNIX and C language. \$3200/mo. Job site & interview in City of Industry. Send resume to 17431 E. Gale Ave., City of Industry, CA 91748.

THE UNIVERSITY OF MICHIGAN
Department of Electrical Engineering
and Computer Science

The Department of Electrical Engineering and Computer Science at The University of Michigan invites applications for positions at all levels in its Computer Science and Engineering Division.

Our emphasis is on the areas of operating systems, distributed systems and networks, database systems, programming languages, software engineering, and theoretical computer science. Exceptional candidates in other areas of computer science and engineering will also be considered. All candidates who apply should have an interest in teaching and a strong research orientation.

Send your resume and the names of at least three references to Professor Toby Teorey, Chair of the Faculty Search Committee, CSE Division, EECS Department, The University of Michigan, Ann Arbor, MI 48109-2122.

The University of Michigan is an Equal Opportunity/Affirmative Action Employer.

SYSTEMS ANALYST

Design data collection and analysis system, coordinate with management and end-users in determining application process and requirements, define requirements and specifications for system development, assist in isolating redundant processes and problems prior to implementation, test, maintain and enhance applications being used. Required Master in Computer Applications with 0 experience. Must be familiar with Lotus 1-2-3 and DBASE IV. \$3,200/mo. Job site & interview in Vernon. Send resume to Personnel Dept. 1912 E. Vernon Ave., Vernon, CA 90058.

SYSTEMS ANALYST

Applicant will perform systems' analysis and software engineering and design with regards to the conversion of the company's database and supporting software programs from Pick and Unix Operation Systems to VAX/VMS Operating Systems. Will also coordinate as project leader 5-7 other programmers. Will utilize knowledge of C Language, VMS Operating System, Unix Operating System, Pick Database.

Minimum requirements: 5 years of college, MS in Computer Science, 1 year experience in job offered, and working knowledge of Basic.

Mon. - Fri. 8.00 a.m. - 5.00 p.m.

\$45,000/yr.

Apply by resume no later than October 31, 1991 to:

Colorado Department of Labor

and Employment

600 Grant Street, Suite 900

Denver, Colorado 80203-3528

Refer to Job Order No.: CO3773842

Attn: Jim Shimada

COMPUTER PROJECT SCIENCE ENGINEER

Minimum 1 year experience with BS degree computer engineering or will accept entry level with MS degree same field. Will be assigned a broad range of projects in connection with our computer development efforts. Duties will include systems research programming, training and support, together with systems research and development, system documentation and technical standard development, working with PRO-IV technology. Salary \$39,400/year. Send resume to EPC International, 4000 MacArthur Blvd., Suite 4400, Newport Beach, California 92660, ATTN: George Kiziroglu, Vice President.

DUKE UNIVERSITY Department of Electrical Engineering

The Department of Electrical Engineering at Duke University seeks an experienced candidate for a tenure track or tenured faculty position in the area of computer engineering. The applicant should have a Ph.D. a strong and documented research record, and a dedication to excellence in teaching. We are particularly interested in a person with interests in fault-tolerance and testability, high performance computing, computer networks or VLSI design and CAD tools. Interested persons should send a curriculum vitae and their names, addresses and phone numbers of five references to: Kishor Trivedi, Computer Engineering Search Committee, Department of Electrical Engineering, Duke University, Durham, NC 27706. Duke University is an equal opportunity/affirmative action employer.

SENIOR SALES ENGINEER LATIN AMERICA

Requires bachelor's degree in Electrical Engineering or Computer Science; 2 years experience in job offered or in computer-related applications/systems engineering; thorough knowledge of and experience with high speed laser printer controller design and sales support; and fluency in technical English and Spanish. Duties include responsibility for sales, supervision, technical support and problem-solving for Company's Mexican, Central, and South American sales network, functioning as technical liaison between U.S. home office and Latin American clients; makes sales calls on management representatives including engineering and other technical personnel at Latin American commercial and industrial businesses; liaison with clients to review various technical equipment and documents to analyze needs and demonstrate desirability and usefulness of company products; provides technical services to clients relating to product use and

operation, including technical training of clients' employees; writes Spanish training manuals for use by clients' technical personnel; confers with clients to design and develop individualized engineering changes for incorporation into company products to improve specific suitability for clients' use; translates clients' programs for company's Engineering, R&D and Marketing Departments for problem solving and implementation of design changes; maintains ongoing training program for Latin American sales distribution network, assist in formulating Latin American business strategy; travels up to 20%, \$40,000/yr. 40 hrs/wk. EEO. Apply or send resume to: D. Abernathy, Alabama State Employment Service, 4130-C Government Blvd., P.O. Box 190399, Mobile, AL 36619 Refer to "AL 0625373.

SENIOR DESIGN ENGINEER

Senior Design Engineer needed to design and develop local area signalling services based switch for telecommunications company. Conduct high level design for call processing switch. Design, implement and test software features for digital telecommunications switch. Provide team leadership for the Bulk Calling Line Identification and switch based visual services using inband technology projects. Monitor completion of activities and assume primary contact position for project issues. Provide ideas and suggestions for future project enhancements. Requires a Bachelor's degree in Computer Science/Computer Engineering and three years experience in job offered or three years directly related digital telecommunications experience involving software development. Experience must include at least one year experience in Pascal as well as End Office switch feature development, trunkline experience and Tandem switch experience. 40 hour work week, \$49,000-72,000 per year. Job Order #NC 3010193, DOT Code 020.061-640. Apply at the Job Service, 516 North Mengum Street, Durham, North Carolina 27701, or your nearest Job Service Office. Ad Paid by an Equal Employment Opportunity Employer.

THE UNIVERSITY OF TENNESSEE SPACE INSTITUTE Faculty Positions in Computer Science

Applications are invited for two tenure-track positions, beginning January or August, 1992, at the rank of assistant professor through professor, in any of these areas: parallel processing, scientific computing, scientific visualization, algorithms, networks, compilers, operating systems, software engineering, and artificial intelligence. A Ph.D. in computer science or a closely related area and demonstrated research potential are required for an assistant professorship; teaching experience is desirable. An established record of quality re-

search and teaching is required for higher ranks.

The computer science program at UTSI emphasizes small, high-quality courses at the graduate level. The load is quite reasonable and allows ample time to develop a strong research program.

UTSI is a multidisciplinary research institute offering degree programs in engineering, physics, applied mathematics and computer science. Computing facilities include an Alliant FX/8, two VAX 11/785 computers, two Symbolics 3670 Lisp machines, several PCs and workstations, and linkage to computers at The University of Tennessee, Knoxville.

Salary is open and commensurate with qualifications. Fringe benefits include group life and medical insurance, TIAA/CREF, and reduced tuition for dependents. UTSI occupies a wooded 365 acre lakeshore campus.

Send a detailed resume and names of three references to: Bruce Whitehead, Chair, Computer Science Search Committee, MS 15, The University of Tennessee Space Institute, Tullahoma, TN 37388-8897 (Phone: (615) 455-0631).

UTSI is an EEO/AA/Title IX/Sect 504/ADA employer.

SOFTWARE ENGINEER/CONSULTANT

Analyze clients' needs to determine the data processing system that will provide system capabilities required for projects or workloads. Analyze, design, program and implement design automation software for a semiconductor manufacturing operation. Develop, and maintain advanced software operating system to maximize the production cell using ANSI C. Test, evaluate and verify systems' reliability, performance, fault tolerance and life cycle. Prepare written summaries supports and recommendations and participate in the development of customer's proposals. Analyze, review, and rewrite programs to increase operating efficiency or adapt to new business procedures. Plan and prepare technical reports, memoranda, and instructional manuals relative to the operation of the program of system. Requires a Bachelor's degree in Computer Science or related area and two years experience in job offered or two years related software development in the manufacturing of integrated circuit processes; or will consider a Master's degree in Computer Science or related area in lieu of the Bachelor and two. Experience should include at least one year experience in the following languages: PASCAL, "C", and UNIX. 40 hour work week, \$17.00 per hour. Apply at the Texas Employment Commission, Dallas, Texas, or send resume to the Texas Employment Commission, TEC Building, Austin, Texas 78778, Job Order #6344505 Ad Paid by an Equal Opportunity Employer.

1951-1991

40 YEARS OF SERVICE



IEEE COMPUTER SOCIETY

A member society of the
Institute of Electrical and Electronics Engineers, Inc.**THE FOLLOWING
INFORMATION IS AVAILABLE:**

Contact the Publications Office,
to facilitate handling, please request by number.

- Membership application, student #203, others #202
- Periodicals subscription form for individuals #200
- Periodicals subscription form for organizations #199
- Publications catalog #201
- Compmail electronic mail brochure #194
- Technical committee list/application #197
- Chapters lists, start-up procedures #193
- Student scholarship application #192
- Volunteer leaders/staff directory #196
- IEEE senior member grade application #204
(requires ten years practice and significant performance in five of those ten)

To check membership status or report a change of address, call the IEEE toll-free number, 1-800-678-4333. Direct all other Computer Society related questions to the Publications Office.

PURPOSE

The IEEE Computer Society advances the theory and practice of computer science and engineering, promotes the exchange of technical information among 100,000 members worldwide, and provides a wide range of services to members and nonmembers.

MEMBERSHIP

Members receive the acclaimed monthly magazine *Computer*, discounts, and opportunities to serve (all activities are led by volunteer members). Membership is open to all IEEE members, affiliate society members, and others interested in the computer field.

EXECUTIVE COMMITTEE

President: Duncan H. Lawrie*

University of Illinois
Dept. of Computer Science
1304 W. Springfield
Urbana, IL 61801
(217) 333-3373

President-Elect: Bruce D. Shriver*
Past President: Helen M. Wood*

VP, Standards: Paul L. Borrelli (1st VP)*
VP, Press Activities: Barry W. Johnson (2nd VP)*
VP, Conferences and Tutorials: Laurel V. Kaleda*
VP, Education: Raymond E. Miller*
VP, Membership Activities: Ronald D. Williams*
VP, Publications: Ronald G. Hoelzman*
VP, Technical Activities: Mario R. Barabasi*

Secretary: James H. Aylor*
Treasurer: Joseph Boykin*
Division V Director: Edward A. Parrish, Jr.*
Division VIII Director: Helen M. Wood*
Executive Director: T. Michael Elliott*

*voting member of the Board of Governors
*nonvoting member of the Board of Governors

BOARD OF GOVERNORS**Term Expiring 1991:**

P. Bruce Bera, Michael Evangelist,
Ted Lewis, Raymond E. Miller, Earl E. Swartzlander, Jr.,
Joseph E. Urban, Thomas W. Williams

Term Expiring 1992:

James H. Aylor, Alvin J. Ellis, Tadao Ichikawa,
C. V. Ramamoorthy, Sallie V. Sheppard,
Harold Stone, Akhiko Yamada

Term Expiring 1993:

Florence Albert-Howard, Jon T. Butler, Michael C. Mulder,
Yale N. Patt, Anneliese von Mayhauser,
Benjamin W. Wah, Ronald Waxman

Next Board Meeting

November 1, 1991, 8:30 a.m.
Oryland Hotel, Nashville, TN

SENIOR STAFF

Executive Director: T. Michael Elliott
Publisher: H. True Seaborn
Director, Conferences and Tutorials: Anne Marie Kelly
Director, Finance and Information Services: Tod S. Hester
Director, Board and Administrative Services: Violet S. Doan
Assistant to the Executive Director: Sandra K. Plau

COMPUTER SOCIETY OFFICES**Headquarters Office**

1730 Massachusetts Ave. NW
Washington, DC 20036-1903
Phone: (202) 371-0101
Fax: (202) 728-9614

Publications Office

10662 Los Vaqueros Cir.
PO Box 3014
Los Alamitos, CA 90720-1264
Membership and General Information:
(714) 821-8380
Publication Orders: (800) 272-6657
Fax: (714) 821-4010

European Office

13, Ave. de L'Acquisition
8-1200 Brussels, Belgium
Phone: 32 (2) 770-21-98
Fax: 32 (2) 770-85-05

Asian Office

Ooshima Building
2-19-1 Minami-Aoyama, Minato-ku
Tokyo 107, Japan
Phone: 81 (3) 3406-3118
Fax: 81 (3) 3409-3553

**IEEE OFFICERS**

President: Eric E. Sumner
President-Elect: Merrill W. Buckley, Jr.
Past President: Carleton A. Bayless
Secretary: Hugh Rudnick
Treasurer: Theodore W. Hissey, Jr.

VP, Educational Activities: Richard S. Nichols
VP, Professional Activities: Michael J. Whitelaw
VP, Publication Activities: J. Thomas Cain
VP, Regional Activities: Robert T. H. Alden
VP, Technical Activities: Fernando Aldana

CALL FOR PAPERS

Call for articles and referees for Computer



Computer seeks articles for inclusion in upcoming special issues.

Molecular Computing has been selected as the theme planned for the **November 1992** issue. Manuscripts reporting survey, original research, design and development, and applications of wafer-scale integration technology are sought immediately in the following areas:

- Molecular computing principles
- Materials and devices
- Architecture
- Neuromolecular computing
- Optomolecular electronic designs
- Carbon versus silicon switches

A 300-word abstract must be submitted by **November 15, 1991**. Fourteen copies of the complete manuscript are due by **January 15, 1992**; notification of decisions is set for **May 15, 1992**; and the deadline for submittal of the final version of each manuscript is **July 1, 1992**.

Submissions and questions should be directed to Michael

Conrad, Department of Computer Science, Wayne State University, Detroit, MI 48202, phone (313) 577-0725, secretary (313) 577-2476, fax (313) 577-6868.

Object-Oriented Systems and Applications is the theme planned for the **September 1992** issue. Manuscripts reporting survey, original research, design and development, and object-oriented systems and applications are sought immediately. See p. 139 of the September 1991 issue of *Computer* for complete details.

Fourteen copies of the complete manuscript are due by **November 15, 1991**; notification of decisions is set for **April 1, 1992**; and the deadline for submittal of the final version of each manuscript is **June 1, 1992**.

Submissions and questions should be directed to David C. Rine, Department of Computer Science, School of Information Technology and Engineering, George Mason University, Fairfax, VA 22030, phone (703) 993-1530, e-mail drine@gmuvmx.gmu.edu; or Bharat Bhargava, Computer Science Department, Purdue University, Lafayette, IN 47907, phone (317) 494-6013, e-mail bb@cs.purdue.edu.

For submittal to Computer, manuscripts must not have been previously published or currently submitted for publication elsewhere. Each manuscript should be no more than 32 typewritten, double-spaced, single-sided pages long, including all text, figures, and references. Each of the 14 copies submitted should include a title page that contains the title of the article, the full name(s) and affiliation(s) of the author(s), complete postal and electronic address(es) of all the authors as well as their telephone and fax number(s), a 300-word abstract, and a list of keywords identifying the central issues of the manuscript's contents. The final manuscript should be approximately 8,000 words in length and contain no more than 12 references.

If you are willing to review articles for any special issue, please send a note listing your research interests to Jon T. Butler, editor-in-chief of Computer, at the Department of Electrical and Computer Engineering, Naval Postgraduate School, Code EC/Bu, Monterey, CA 92943-5004, Internet butler@ece.nps.navy.mil.

ICCI 92, Fourth Int'l Conf. on Computing and Information: May 28-30, 1992, Toronto. Submit manuscript to Anestis A. Topitsis, Computer Science, Atkinson College, York Univ., Toronto, Ont., Canada M3J 1P3, (416) 736-5232, fax (416) 736-5773, e-mail anestis@cid.yorku.ca or anestis@yetti.cs.yorku.ca.

VLDB Journal is expected to be published for the first time during the first half of 1992. Submit six copies of paper to Fred J. Maryanski, Univ. of Connecticut, U-155, Rm. 209, 260 Glenbrook Rd., Storrs, CT 06269-3155, phone (203) 486-2421, fax (203) 486-6379, e-mail fred@sparco.brc.uconn.edu.

TOOLS Europe 92, Tech. of Object-Oriented Languages and Systems Int'l Conf.: Mar. 30-Apr. 2, 1992, Dortmund, Germany. Cosponsors: Societe des Outils du Logiciel and Georg Heep. Submit paper by **Oct. 21, 1991**, to Boris Magnusson, Lund Univ. Computer Science Dept., PO Box

118, S-22100 Lund, Sweden, fax (46) 4613-1021, e-mail boris@dna.lth.se; or SOL, 14 rue Jean Rey, 75015 Paris, France, phone 33 (1) 4056-0358, fax 33 (1) 4056-0581.

IEEE Expert plans a special track on artificial intelligence in text-based information systems. Submit four copies of the full manuscript by **Oct. 25, 1991**, to Dirk L. Lee, Computer and Information Science Dept., Ohio State Univ., 2036 Neil Ave., Columbus, OH 43210-1277, e-mail dllee@cis.ohio-state.edu; or W. Bruce Croft,

Computer Science Dept., Univ. of Massachusetts, Lederle Graduate Research Center, Amherst, MA 01003, e-mail croft@cs.umass.edu.

Second Int'l Conf. on Systems Integration: June 15-18, 1992, Morristown, N.J. Cosponsor: ACM et al. Submit six copies of complete paper and abstract by **Oct. 25, 1991**, and final manuscript by **Feb. 17, 1992**, to Raymond T. Yeh, c/o Peter A. Ng, Computer and Information Science Dept., New Jersey Inst. of Tech.,

Computer recently published Call for Papers notices for events whose initial submittal deadlines had expired but whose final-version deadlines were still in effect. Since final-version deadlines apply only to papers that have already been accepted, some conferences received post-deadline enquiries. Among them were the Sixth Workshop on Parallel and Distributed Simulation, the Eighth International Conference and Workshop on Data Engineering, and the 11th International Phoenix Conference on Computers and Communications. Computer apologizes to anyone who might have been inconvenienced.

University Heights, Newark, NJ 07102, phone (201) 596-3387, e-mail ng_p@vienna.njit.edu.

Ⓢ DAC 92, 29th IEEE/ACM Design Automation Conf: June 8-12, 1992, Anaheim, Calif. Submit eight copies of complete paper by **Oct. 25, 1991**, to Bryan Preas, c/o MP Associates, 7490 Clubhouse Dr., Boulder, CO 80301, phone (303) 530-4333.

ICPR 92, 11th Int'l Conf. on Pattern Recognition: Aug. 30-Sept. 3, 1992, The Hague, The Netherlands. Sponsor: Int'l Assoc. for Pattern Recognition. Submit four copies of extended abstract by **Oct. 31, 1991**, to ICPR Secretariat, Delft Univ. of Tech., Electrical Eng. Dept., PO Box 5031, NL-2600 GA Delft, The Netherlands, phone 31 (15) 786-052, fax 31 (15) 622-000, e-mail icpr@et.tudelft.nl.

Vision Interface 92: May 11-15, 1992, Vancouver, B.C., Canada. Sponsor: Canadian Image Processing and Pattern Recognition Soc. Submit five copies of full paper by **Oct. 31, 1991**, to Xiaobo Li, Computing Science Dept., 615 General Services Bldg., Univ. of Alberta, Edmonton, Alta., Canada, T6G 2H1, phone (403) 492-2299, fax (403) 492-1071, e-mail vi92@cs.ualberta.ca.

Graphics Interface 92: May 11-15, 1992, Vancouver, B.C., Canada. Sponsor: Canadian Human-Computer Comm. Soc. Submit five copies of full paper by **Oct. 31, 1991**, and final paper by **Mar. 27, 1992**, to Eugene Fiume, Computer Science Dept., Univ. of Toronto, Toronto, Ont., Canada M5S 1A4, phone (416) 978-5472, e-mail elf@dgp.toronto.edu.

Ⓢ Comp Euro 92, IEEE Int'l Conf. on Computer Systems and Software Eng.: May 4-8, 1992, The Hague, The Netherlands. Cosponsors: IEEE Region 8, IEEE Benelux Section. Submit five copies of paper by **Nov. 1, 1991**, and final manuscript by **Feb. 1, 1992**, to Patrick Dewilde, Delft Univ. of Tech., EE Dept., POB 5031, 2600 GA Delft, The Netherlands, phone 31 (15) 785-089, fax 31 (15) 623-271.

Ⓢ IEA/AIE 92, 25th Int'l Conf. on Industrial and Eng. Applications of Artificial Intelligence and Expert Systems: June 9-12, 1992, Paderborn, Germany. Cosponsors: Univ. of Paderborn et al. Submit six copies of paper by **Nov. 1, 1991**, and final copy by **Mar. 16, 1992**, to Fevzi Belli, Univ. of Paderborn, Postfach 1621, 4790 Paderborn, Germany, phone 49 (5251) 603-282, fax 49 (5251) 602-519, e-mail bellif@adt.uni-paderborn.de.

Ⓢ 22nd Int'l Symp. on Multiple-Valued Logic: May 27-29, 1992, Sendai, Japan. Sponsors: IEEE Computer Soc. Technical Committee on Multiple-Valued Logic, Japan Research Group on Multiple-Valued Logic. Submit five copies of paper by **Nov. 1, 1991**, and camera-ready copy by **Mar. 1, 1992**, to K.C. Smith, Electrical Eng. Dept., Univ. of Toronto, Toronto, Ont.,

Canada M5S 1A4, phone (416) 978-5033, fax (416) 978-7423, e-mail lfujino@csri.toronto.edu (in the Americas); P.W. Besslich, FB-1, Electrical Eng. Dept., Univ. of Bremen, PO Box 330-440, D-2800 Bremen 33, Germany, phone 49 (421) 218-3526, fax 49 (421) 218-3601 (in Europe and Africa); or T. Sasao, Computer Science and Electronics Dept., Kyushu Inst. of Tech., Iizuka 820, Japan, phone 81 (948) 29-7675, fax 81 (948) 29-7651, e-mail sasao@aries.cse.kyutech.ac.jp (in Asia and the Pacific).

Ⓢ ISCA 92, 19th Int'l Symp. on Computer Architecture: May 19-21, 1992, Gold Coast, Queensland, Australia. Cosponsors: IEEE, ACM SIGARCH. Submit five copies of manuscript by **Nov. 1, 1991**, and camera-ready copy by **Mar. 6, 1992**, to Allan Gottlieb, NYU Ultracomputer Lab, 715 Broadway, 10th Floor, New York, NY 10003, e-mail gottlieb@nyu.edu.

Ⓢ CVPR 92, IEEE Computer Society Conf. on Computer Vision and Pattern Recognition: June 15-18, 1992, Champaign, Ill. Submit four copies of complete manuscript by **Nov. 1, 1991**, to Sharon Collins, Univ. of Illinois, Beckman Inst., 405 N. Mathews, Urbana, IL 61801.

DIAC 92, Directions and Implications of Advanced Computing: May 2-3, 1992, Berkeley, Calif. Sponsor: Computer Professional Society for Social Responsibility. Submit four copies of paper by **Nov. 1, 1991**, and camera-ready version by **Mar. 1, 1992**, to Douglas Schuler, Boeing Computer Services, MS 7L-64, PO 24346, Seattle, WA 98124-0346, home phone (206) 632-1659, office phone (206) 865-3832, fax (206) 865-2964, e-mail dschuler@june.cs.washington.edu.

Coling 92, 14th Int'l Conf. on Computational Linguistics: July 23-28, 1992, Nantes, France. Sponsor: Int'l Committee on Computational Linguistics. Submit six copies of full paper by **Nov. 1, 1991**, and camera-ready version by **May 1, 1992**, to A. Zampolli, Univ. di Pisa, ILC, via della Faggiola 32, I-56100, Pisa, Italy, phone 39 (50) 560-481, fax 39 (50) 589-055.

SEEDS III, Third Symp. on Experiences with Distributed and Multiprocessor Systems: Mar. 26-27, 1992, Newport Beach, Calif. Sponsor: Usenix Assoc. Submit six copies of full paper by **Nov. 1, 1991**, and camera-ready copy by **Jan. 24, 1992**, to Gene Spafford, Software Eng. Research Center, Computer Science Dept., Purdue Univ., West Lafayette, IN 47907-1398, phone (317) 494-7825, e-mail spaf@cs.purdue.edu.

11th Int'l Workshop on Distributed Artificial Intelligence: Feb. 26-28, 1992, Glen Arbor, Mich. Cosponsors: Univ. of Michigan, Industrial Tech. Inst. Submit four copies of paper by **Nov. 1, 1991**, to Edmund H. Durfee, Electrical Eng. and Computer Science Dept., Univ. of Michigan, Ann Arbor, MI 48109, phone (313) 936-1563, fax (313) 763-1260, e-mail durfee@caen.engin.umich.edu.

NATUG, North Am. Transputer Users Group 1992 Spring Meeting: Apr. 19-21, 1992, Baltimore, Md. Submit 1,000-1,500-word abstract by **Nov. 1, 1991**, and final draft of accepted paper by **Jan. 31, 1992**, to Andrew Veronis, Univ. of Maryland, Eastern Shore, Computer Science Dept., Prince Anne, MD 21853, phone (301) 651-2000, e-mail averonis@umes.umd.edu.

1991 Workshop on Architectural Aspects of Real-Time Systems: Dec. 3, 1991, San Antonio, Texas. Sponsor: IEEE. Submit four copies of extended summary by **Nov. 3, 1991**, to C.M. Krishna, Electrical and Computer Eng. Dept., Univ. of Massachusetts, Amherst, MA 01003, phone (413) 545-0766, e-mail krishna@ecs.umass.edu.

Ⓢ CGI 92, Computer Graphics Int'l Conf.: June 22-26, 1992, Tokyo. Cosponsors: IEEE et al. Submit six copies of full paper by **Nov. 8, 1991**, to T.L. Kunii, Information Science Dept., Univ. of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113, Japan, phone 81 (3) 3812-2111 ext. 4116, fax 81 (3) 3818-4607.

Ⓢ 1992 IEEE Symp. on Research in Security and Privacy: May 4-6, 1992, Oakland, Calif. Sponsor: IEEE Computer Soc. Technical Committee on Security and Privacy. Submit six copies of paper by **Nov. 8, 1991**, and camera-ready version by **Feb. 28, 1992**, to John McLean, Naval Research Lab., Code 5543, Washington, DC 20375, phone (202) 767-3852, e-mail mclea@nrl.nrl.navy.mil.

Ⓢ DCC 92, Data Compression Conf.: Mar. 24-26, 1992, Snowbird, Utah. Sponsors: IEEE Computer Soc. Technical Committee on Computer Comm., NASA. Submit four copies of extended abstract by **Nov. 15, 1991**, and camera-ready draft by **Jan. 13, 1992**, to Martin Cohn, Computer Science Dept., Brandeis Univ., Waltham, MA 02254, (617) 736-2705, fax (617) 736-2741, e-mail marty@cs.brandeis.edu.

Ⓢ Parle 92, Parallel Architectures and Languages Europe: June 15-18, 1992, Paris. Cosponsors: IEEE Region 8 et al. Submit eight copies of full draft not exceeding 6,000 words by **Nov. 15, 1991**, to Daniel Etienne, LRI, Bat 490, Univ. of Paris Sud, 91405 Orsay Cedex, France, phone 33 (1) 6941-6621, fax 33 (1) 6941-6586, e-mail de@lri.lri.fr.

Int'l J. of Computers and Electrical Eng. plans a special issue on parallel and distributed computing for intelligent systems. Submit manuscript by **Nov. 15, 1991**, to Nageswara S.V. Rao, Computer Science Dept., Old Dominion Univ., Norfolk, VA 23529-0162, phone (804) 683-4414, fax (804) 683-4900, e-mail rao@cs.odu.edu.

Int'l J. of Software Eng. and Knowledge Eng. plans a special issue on domain analysis for software engineering. Submit five copies of complete manuscript by **Nov. 15, 1991**, to Guillermo Arango, Schlumberger

Lab for Computer Science, 8311 North FM 620, Austin, TX 78720, phone (512) 331-3735, e-mail arango@scs.slb.com; or Bruce I. Blum, Applied Physics Lab, Johns Hopkins Rd., Laurel, MD 20723-6099, phone (301) 953-6235, e-mail bib@aplcomm.jhuapl.edu.

Seventh Symp. on Microcomputer and Microprocessor Applications: Apr. 22-24, 1992, Budapest, Hungary. Cosponsors: Scientific Soc. for Telecommunication et al. Submit four copies of paper by **Nov. 15, 1991**, and camera-ready version by **Feb. 15, 1992**, to SST, Budapest 5, P.L. 451, Hungary, H-1372, phone (361) 153-1027, fax (361) 153-0451.

SPHCC 92, Scalable High-Performance Computing Conf.: Apr. 26-29, 1992, Williamsburg, Va. Submit extended abstract by **Nov. 15, 1991**, and final paper by **Mar. 1, 1992**, to Emily Todd, ICASE, MS 312-C, NASA Langley Research Center, Hampton, VA 23665, e-mail emily@icase.edu.

SID 92, Int'l Symp. and Seminar of the Soc. for Information Display: May 17-22, 1992, Boston. Submit abstract and technical summary by **Nov. 15, 1991**, and four-page camera-ready copy by **Mar. 16, 1992**, to Paul M. Alt, IBM T.J. Watson Research Center, PO Box 218, Yorktown Heights, NY 10598, phone (914) 945-2437, fax (914) 945-1974, e-mail alt@watson.ibm.com; Joyce Farrell, Hewlett-Packard Labs, PO Box 10490, Palo Alto, CA 94303-0969, phone (415) 857-2807, fax (415) 857-4320, e-mail farrell@hplabs.hp.com; or Jay Morreale, Palisades Inst. for Research Services, 201 Varick St., Suite 1140, New York, NY 10014, phone (212) 620-3371, fax (212) 620-3379.

IEEE Design & Test seeks papers (25 double-spaced pages, including figures) on general design, test, and manufacturing topics by **Dec. 1, 1991**, to Manuel d'Abreu, GE R&D, 1 River Rd., PO Box 8, KW-C308, Schenectady, NY 12301, phone (518) 387-7097, fax (518) 387-5299, e-mail dabrau@crd.ge.com.

Third Workshop on Workstation Operating Systems: Apr. 23-24, 1992, Key Biscayne, Fla. Sponsor: IEEE Computer Soc. Technical Committee on Operating Systems and Applications Environments. Submit eight copies of position statement by **Dec. 1, 1991**, to Anita Borg, Western Research Lab, Digital Equipment, 250 University Ave., Palo Alto, CA 94301, (415) 617-3315, e-mail borg@wrl.dec.com.

ITS 92, Second Int'l Conf. on Intelligent Tutoring Systems: June 10-12, 1992, Montreal. Cosponsors: Univ. of Montreal et al. Submit five copies of manuscript (up to 5,000 words) by **Dec. 1, 1991**, to Claude Frasson, Département d'IRO, Université de Montréal, CP 6128 Succ. A, Montréal, Que., Canada H3C 3J7, phone (514) 343-7019, fax (514) 343-5834, e-mail frasson@iro.umontreal.ca.

CALENDAR



In the accompanying Calendar and adjoining Call for Papers, the IEEE Computer Society logo identifies the conferences the society is participating in or sponsoring. Other conferences of interest to our readers — as well as their sponsors — are also listed.

Calendar and Call for Papers notices are published free of charge, in chronological order, and as space permits.

When submitting information for inclusion in the Call for Papers section, please provide the following: the name of the event, the date(s), the location, the sponsor(s), the deadline for submittals, and the phone and fax numbers and — if applicable — the electronic address of the person to whom papers should be sent.

For listings in the Calendar section, please provide the following: the event name, the date(s), the location, the sponsor(s), and the phone and fax numbers and — again, if applicable — the electronic address of the person to contact for complete information.

Submitted information should arrive at **Computer at least five weeks before the month of publication** (i.e., for the **December 1991** issue, send information for receipt by **October 21, 1991**) to Chuck Governale, Calendar Dept., Computer, PO Box 3014, Los Alamitos, CA 90720-1264, fax (714) 821-4010, e-mail c.governale@compmail.com.

October 1991

ICSP 1, First Int'l Conf. on the Software Process, Oct. 21-22, Redondo Beach, Calif. Contact ICSP 1, Rocky Mountain Inst. of Eng., PO Box 3521, Boulder, CO 80303, phone (303) 499-4782, e-mail icsp1@cs.sda.com.

SimTec 91, Int'l Conf. on Simulation Tech., Oct. 21-23, Orlando, Fla. Sponsor: Soc. for Computer Simulation. Contact SCS, PO Box 17900, San Diego, CA 92177-7900, phone (619) 277-3888, fax (619) 277-3930.

VDM 91, Fourth VDM Europe Symp. on Formal Software Development Methods, Oct. 21-25, Noordwijkerhout, The Netherlands. Contact Hans Toetenel, Delft Univ. of Tech., Computer Science Dept., PO Box 356, 2600 AJ Delft, The Netherlands, phone 31 (15) 782-518, fax 31 (15) 787-141, e-mail toet@dtuab.tudelft.nl.

Visualization 91, Oct. 22-25, San Diego, Calif. Sponsor: IEEE Computer Soc. Technical Committee on Computer Graphics. Contact Bruce Brown, Oracle, 500 Oracle Pkwy., MD 40P12, Redwood Shores, CA 94065, phone (415) 726-0983, fax (415) 506-7200; or Gregory M. Nielson, Computer Science Dept., Arizona State Univ., Rural Road and University, Tempe, AZ 85287-5406, phone (602) 965-2785.

10th Int'l Conf. on Entity-Relationship Approach, Oct. 23-25, San Francisco. Sponsor: ER Inst. et al. Contact Terry Moriarty, Bank of Am., PO Box 37000, #3439, San Francisco, CA 94137, phone

(415) 675-5250; or Toby J. Teorey, EECS Dept., Univ. of Michigan, Ann Arbor, MI 48109-2122, phone (313) 765-5216, e-mail teorey@ub.cs.umich.edu.

Sixth Int'l Workshop on Software Specification and Design, Oct. 25-26, Como, Italy. Contact C. Ghezzi, Dip. di Elettrotecnica, Politecnico di Milano, Piazza Leonardo Da Vinci 32, 20133 Milano, Italia, e-mail relet24@impoli.bitnet; or Jean-Pierre Finance, CRIN, Campus Scientifique, BP 239 54000 Nancy, France, e-mail finance@loria.crin.fr.

1991 Lasers in Graphics/Electronic Design in Print Conf., Oct. 26-31, Ft. Lauderdale, Fla. Contact Patrice M. Dunn, c/o Dunn Technology, 1855 E. Vista Way, Vista, CA 92084, fax (619) 758-5401.

ITC 91, Int'l Test Conf., Oct. 26-Nov. 1, Nashville, Tenn. Sponsor: IEEE Philadelphia Section. Contact Doris Thomas, PO Box 264, Mt. Freedom, NY 07970, phone (201) 895-5260, fax (201) 896-7265; or IEEE Computer Soc., 1730 Massachusetts Ave. NW, Washington, DC 20036-1903, phone (202) 371-1013.

VHDL Int'l Users' Forum, Oct. 27-30, Newport Beach, Calif. Cosponsors: VHDL Int'l, Electronic Industries Assoc. Contact Conf. Management Services, 407 Chester St., Menlo Park, CA 94025, phone (415) 329-0510, fax (415) 324-3150.

ILPS 91, Int'l Logic Programming Symp., Oct. 28-31, San Diego, Calif. Sponsor: Assoc. of Logic Programming. Contact Kenneth Kahn or Vijay Saraswat, Xerox PARC, 3333 Coyote Hill Rd., Palo

Alto, CA 94304, fax (415) 494-4334, e-mail ilips91@parc.xerox.com, Kahn's phone (415) 494-4390, Saraswat's phone (415) 494-4747.

Eurographics Workshop on Computer Graphics and Math, Oct. 28-31, Genova, Italy. Sponsor: Inst. for Applied Math, Italian Nat'l Research Council, Contact Bianca Falcidieno or Caterina Pienovi, Istituto per la Matematica Applicata, Via L.B. Alberti 4, 16132 Genova, Italy, phone 39 (10) 517-639, fax 39 (10) 517-801, e-mail falcidieno@image.ge.cnrit.it or pienovi@image.ge.cnrit.it.

Iecon 91, 17th Int'l Conf. on Industrial Electronics, Control, and Instrumentation, Oct. 28-Nov. 1, Kobe, Japan. Contact Allen C. Chen, AT&T Bell Labs, Rm. 1E134A, Whippany Rd., PO Box 903, Whippany Road, NJ 07981-0903, or Hiro Haneda, Electronics Eng. Dept., Kobe Univ., Rokko-dai, Nada-ku, Kobe City, Hyogo 857, Japan, phone 81 (78) 3881-1212, fax 81 (78) 3861-7879.

Congress of the Italian Assoc. for Artificial Intelligence, Oct. 29-31, Palermo, Italy. Contact Salvatore Gagliano, Centro per la Ricerca Elettronica in Sicilia, Viale Regione Siciliana, 49, 90046, Monreale, Palermo, Italy.

ISCIS VI, Sixth Int'l Symp. on Computer and Information Sciences, Oct. 30-Nov. 2, Ankara, Turkey. Sponsors: Bilkent Univ., IEEE Turkey Chapter, Contact Mehmet Baray, Bilkent Univ., Computer Eng. and Information Sciences Dept., Bilkent 06533 Ankara, Turkey, phone (90) 4-266-4133, fax (90) 4-266-4127, e-mail iscis@trbilun.bitnet.

November 1991

Fifth Southeastern Small-College Computing Conf., Nov. 1-2, Nashville, Tenn. Sponsor: Consortium for Computing in Small Colleges. Contact Frank Cheatham, Campbellsville College, 200 W. College St., Campbellsville, KY 42718, phone (502) 465-8158.

User Services Conf. XIX, Nov. 3-6, Seattle. Sponsor: ACM. Contact Sheryl Burgstahler, Univ. of Washington HG-45, Seattle, WA 98195, phone (206) 543-0622, e-mail sheryl@cac.washington.edu.

Tech. Conf. on Optical Pattern Recognition, Nov. 3-8, San Jose, Calif. Sponsor: Int'l Soc. for Optical Eng. Contact Robert Hart, SPIE, PO Box 10, Bellingham, WA 98227-0010, phone (206) 676-3290, fax (206) 647-1445.

25th Asilomar Conf. on Signals, Systems, and Computers, Nov. 4-6, Pacific Grove, Calif. Sponsors: Naval Postgraduate School, San Jose State Univ., Contact Frederic J. Harris, Electrical and Computer Eng. Dept., San Jose State

Univ., San Diego, CA 92182-0190, (619) 594-6162.

Gomac '91, Government Microcircuit Applications Conf., Nov. 5-7, Kissimmee, Fla. Cosponsors: US Dept. of Defense et al. Contact Walter J. Schoppe, Gomac '91, Naval Air Development Center, Code 40F Warminster, PA 18974-6000, phone (216) 441-2378.

COCS 91, Conf. on Organizational Computing Systems, Nov. 5-8, Atlanta. Sponsors: IEEE Computer Soc. Technical Committee on Office Automation, ACM SIGOIS, Int'l Federation for Information Processing. Contact Frederick H. Lochovsky, Hong Kong Univ. of Science and Tech., Computer Science Dept., 5/F World Shipping Center, 7 Canton Rd., Hong Kong, phone (852) 302-1642, fax (852) 736-8801, e-mail csfred@usthk.bitnet or csfred@uxmail.ust.hk (Internet).

SERF, First Software Eng. Research Forum, Nov. 7-8, Tampa, Fla. Contact Chuck Ingle, CS Dept., F.I.T., Melbourne, FL 32901, e-mail cingle@cs.fit.edu.

TAI 91, Third IEEE Computer Soc. Conf. on Tools for Artificial Intelligence, Nov. 10-13, San Jose, Calif. Contact Benjamin Wah, Coordinated Science Lab, MC 228, Univ. of Illinois, 1101 W. Springfield Ave., Urbana, IL 61801-3082, phone (217) 333-3516, fax (217) 244-1764, e-mail wah%aquinas@csu.uiuc.edu; or Nikolaus G. Bourbakis, 4138 Moonflower Ct., San Jose, CA 95135, phone (408) 270-3455.

Sixth Conf. on Visual Comm. and Image Processing, Nov. 10-13, Boston. Sponsors: Int'l Soc. for Optical Eng. et al. Contact SPIE, PO Box 10, Bellingham, WA 98227-0010, phone (206) 676-3290.

ICCADC 91, IEEE Int'l Conf. on Computer-Aided Design, Nov. 10-14, Santa Clara, Calif. Sponsor: IEEE Circuits and Systems Soc. Contact ICCADC 91 Secretary, MP Associates, 7490 Clubhouse Dr., Suite 102, Boulder, CO 80301, phone (303) 530-4562.

UIST 91, Fourth Symp. on User Interface Software and Tech., Nov. 11-13, Hilton Head, S.C. Cosponsors: ACM Siggraph and SIGCHI. Contact Joek Mackinlay, Xerox PARC, 3333 Coyote Hill Rd., Palo Alto, CA 94304, phone (415) 494-4335, fax (415) 494-4777, e-mail mackinlay.parc@xerox.com.

ASM 91, Second Int'l Conf. on Applications of Software Measurement, Nov. 11-14, Orlando, Fla. Cosponsors: Data Processing Management Assoc. Contact Linda Ross, 2754 Quentin Ave. South, Minneapolis, MN 55416.

Fourth Int'l Conf. on Networks and Electronic Office Systems, Nov. 12-14, London. Sponsor: Institution of Electrical Engineers. Contact Conf. Services, IEE, Savoy Place, London WC2R 0BL, UK, phone 44 (71) 240-1871, fax 44 (71) 240-7735.

Fourth Int'l Symp. on Artificial Intelligence, Nov. 13-15, Cancun, Mexico. Sponsor: Inst. Tech. de Estudios Superiores de Monterrey. Contact Hugo Terashima, Centro de Inteligencia Artificial, ITESM, Ave. Eugenio Garza Sada No. 2501, Col. Tecnológico, CP 64849 Monterrey, N.L. Mexico, phone 52 (83) 58-2000 ext. 5134, fax 52 (83) 58-1400, e-mail isai@tecmytm.bitnet.

Supercomputing Debugging Workshop 91, Nov. 14-16, Albuquerque, N.M. Sponsors: Los Alamos Nat'l Lab. Lawrence Livermore Nat'l Lab. Contact Jeffrey S. Brown, LANL, Los Alamos, NM 87545, phone (505) 665-4655, e-mail jxyb@lanl.gov.

SCAN 91, 11th Symp. on Small Computers in the Arts, Nov. 15-17, Philadelphia. Sponsor: Small Computers in the Arts Network. Contact Misako Scott, 5132 Hazel Ave., Philadelphia, PA 19143, phone (215) 472-2392.

Second Int'l Workshop on Network and Operating System Support for Digital, Audio, and Video, Nov. 18-19, Heidelberg, Germany. Sponsor: ACM. Contact Ralf G. Hertwich, IBM Tiergartenstr. 8, D-6900 Heidelberg, Germany, phone 49 (6221) 404-218, e-mail rgh@hdbml.bitnet.

Micro 24, 24th ACM/IEEE Int'l Symp. and Workshop on Microarchitecture, Nov. 18-20, Albuquerque, N.M. Cosponsor: ACM. Contact Yashwant K. Malaiya, Computer Science Dept., Colorado State Univ., Fort Collins, CO 80523, phone (303) 491-7031, fax (303) 491-2293, e-mail malaiya@ravi.cs.colostate.edu.

1991 IEEE Int'l Workshop on Defect and Fault Tolerance in VLSI Systems, Nov. 18-20, Hidden Valley, Pa. Contact Wojciech Malý, Electrical and Computer Eng. Dept., Carnegie Mellon Univ., 5000 Forbes Ave., Pittsburgh, PA 15213-2890, phone (412) 268-6637, fax (412) 268-2860, e-mail maly@ece.cmu.edu.

ANN 91, Second Int'l Conf. on Artificial Neural Networks, Nov. 18-20, Bournemouth, UK. Sponsor: Institution of Electrical Engineers. Contact IEE Conf. Services, Savoy Place, London WC2R 0BL, UK, phone (071) 240-1871, fax (071) 240-7735.

IJCNN 91 Singapore, Int'l Joint Conf. on Neural Networks, Nov. 18-21, Singapore. Cosponsors: IEEE Neural Networks Council, Int'l Neural Networks Soc. Contact Noml Feldman, Meeting Management, 5565 Oberlin Dr., Suite 110, San Diego, CA 92121, fax (619) 535-3880, Toshio Fukuda, IJCNN 91 Singapore, Mechanical Eng. Dept., Nagoya Univ., Furo-cho, Chikusa-ku, Nagoya 464-01, Japan, fax 81 (52) 3781-9243, or Teck-Seng Low, IJCNN 91 Singapore, Comm. Int'l Associates Pte. Ltd., 44/46 Tanjong Pagar Rd., Singapore 0208, fax (65) 226-2877.

Supercomputing 91, Nov. 18-22, Albuquerque, N.M. Cosponsor: ACM. Contact Raymond L. Elliott, Computing and Comm. Division, MS B260, Los Alam-

os Nat'l Lab, Los Alamos, NM 87545, phone (505) 667-1449, fax (505) 665-4361, e-mail rle@lanl.gov; or Supercomputing 91, IEEE Computer Soc., 1730 Massachusetts Ave. NW, Washington, DC 20036-1903, phone (202) 371-1013.

Ⓛ Fifth Workshop on Software Reuse, Nov. 18-22, Reston, Va. Contact William J. Tracz, IBM—MD 0210, Owego, NY 13827, phone (607) 751-2109, fax (607) 751-2169.

December 1991

Ⓛ Third IEEE Int'l Symp. on Parallel and Distributed Processing, Dec. 1-5, Dallas. Sponsors: IEEE Computer Soc., IEEE Dallas Section, ACM SIGARCH. Contact Sartaj Sahni, CIS Dept., CSE 301, Univ. of Florida, Gainesville, FL 32611, phone (904) 392-1527, fax (904) 392-1220; or Behrooz Shirazi, Univ. of Texas at Arlington, Computer Science Eng. Dept., Box 19015, Arlington, TX 76019-0015, phone (817) 273-3605, fax (817) 273-2548, e-mail shirazi@evax.utarl.edu.

Ⓛ Fourth Int'l Workshop on Petri Nets and Performance Models, Dec. 3-5, Melbourne, Australia. Sponsor: Telecom Australia. Contact Jonathan Billington, Telecom Australia Research Labs, PO Box 249, 770 Blackburn Rd., Clayton North, Vic. 3168, Australia, phone (61) 541-6416, fax (61) 544-2362, e-mail j.billington@trlz.oz.au.

Ⓛ 12th IEEE Symp. on Real-Time Systems, Dec. 3-6, San Antonio, Texas. Sponsor: IEEE Computer Soc. Technical Committee on Real-Time Computing. Contact Jane S.W. Liu, Computer Science Dept., Univ. of Illinois, 1304 W. Springfield Ave., Urbana, IL 61801, phone (217) 333-0135, e-mail janeliu@cs.uiuc.edu.

Ⓛ PDIS, First Int'l Conf. on Parallel and Distributed Information Systems, Dec. 4-6, Miami Beach, Fla. Cosponsors: IEEE Computer Soc. et al. Contact Amit Sheth, Bellcore, 1J-210, 444 Hoes Ln., Piscataway, NJ 08854, phone (908) 699-3300, fax (908) 699-9011, e-mail amit@ctt.bellcore.com.

Ⓛ WSC91, 1991 Winter Simulation Conf., Dec. 8-11, Phoenix, Ariz. Contact EPIC Management, 8720 Red Oak Blvd., Suite 224, Charlotte, NC 28217, phone (800) 447-6949; or Robert Crain, Wolverine Software, 4115 Annandale Rd., Suite 200, Annandale, VA 22003.

January 1992

Ⓛ Fifth Int'l Conf. on VLSI Design, Jan. 4-7, Bangalore, India. Sponsors: VLSI Soc. of India et al. Contact Asok K. Laha, Cadence Design Systems, Systems Division, 2 Lowell Research Center Dr.,

Lowell, MA 01852-4995, phone (508) 934-0233, fax (508) 441-1109, e-mail laha@cadence.com; or Lalit M. Patnaik, Computer Science and Automation, Indian Inst. of Science, Bangalore, 560012, India, phone 91 (812) 342-451, e-mail lalit%vigyan@shakti.emet.in.

Ⓛ HICSS 25, Hawaii Int'l Conf. on Systems Sciences, Jan. 7-10, Koloa, Hawaii. Cosponsors: IEEE, ACM, Computer Luqi, Computer Science Dept., Naval Postgraduate School, Monterey, CA 93940, phone (408) 646-2468; Ronald J. Norman, IDS Dept., San Diego State Univ., San Diego, CA 92182-0127, phone (619) 594-3734, fax (619) 594-1573, Internet norman@sciences.sdsu.edu; or Ralph Sprague, Univ. of Hawaii at Manoa Decision Science, 2404 Maile Way, Honolulu, HI 96822, phone (808) 956-6606, e-mail hicc@uhcux.uhcc.hawaii.edu.

Ⓛ PADS, Sixth Workshop on Parallel and Distributed Simulation, Jan. 20-22, Newport Beach, Calif. Joint sponsors: IEEE, ACM, Soc. for Computer Simulation. Contact Paul Reynolds, Computer Science Dept., Olsson Hall, Univ. of Virginia, Charlottesville, VA 22903, phone (804) 924-1039, e-mail pfr@louisxiv.ipc.virginia.edu.

Ⓛ IEEE Int'l Conf. on Wafer-Scale Integration, Jan. 22-24, San Francisco. Sponsors: IEEE Computer Soc., IEEE Components, Hybrids, and Manufacturing Tech. Soc. Contact Peter W. Wyatt, MIT Lincoln Lab, B-153, Box 73, Lexington, MA 02173-9108, phone (617) 981-7232.

Ⓛ SETA 2, Second Int'l Symp. on Environments and Tools for Ada, Jan. 28-31, Herndon, Va. Cosponsors: IEEE Computer Soc. Technical Committee on Computer Languages, ACM. Contact Patricia Ornborn, NADC Code 7031, Warminster, PA 18974-5000, phone (215) 441-2737, e-mail tricia@nadc.navy.mil.

February 1992

Ⓛ RIDE 92, Int'l Workshop on Research Issues in Data Eng., Feb. 2-3, Mission Palms, Ariz. Contact Clement Yu, EECS Dept., Univ. of Illinois at Chicago, Chicago, IL 60680, phone (312) 996-2318, fax (312) 413-0024, e-mail yu@uicbert.eecs.uic.edu.

Ⓛ PADSE 92, Eighth Int'l Conf. on Data Eng., Feb. 3-7, Phoenix, Ariz. Sponsor: IEEE Computer Soc. Technical Committee on Data Eng. Contact Nick J. Cercone, Center for Systems Sciences, Simon Fraser Univ., Burnaby, B.C., Canada V5A 1S6, phone (604) 291-4588 or 3229, fax (604) 291-4951, e-mail nick@cs.sfu.ca; or (for workshop on research issues) Clement Yu, Electrical Eng. and Computer Science Dept., Univ. of Illinois at Chicago, Chicago, IL 60680, phone (312) 996-2318, fax (312) 413-0024.

Ⓛ Compcon Spring 92, Feb. 24-28, San Francisco. Contact Compcon Spring 92, IEEE Computer Soc., 1730 Massachusetts Ave. NW, Washington, DC 20036-1903, phone (202) 371-1013.

Ⓛ GLS-VLSI 92, Second Great Lakes Symp. on VLSI, Feb. 28-29, Kalamazoo, Mich. Sponsor: Western Michigan Univ. Contact Naved Sherwan, Computer Science Dept., Western Michigan Univ., Kalamazoo, MI 49008, phone (616) 387-5662, fax (616) 387-3999.

March 1992

Ⓛ CAIA 92, Eighth IEEE Conf. on Artificial Intelligence for Applications, Mar. 2-6, Monterey, Calif. Contact CAIA 92, IEEE Computer Soc., 1730 Massachusetts Ave. NW, Washington, DC 20036-1903, phone (202) 371-1013, fax (202) 728-0884.

Ⓛ Third Int'l Conf. on Data and Knowledge Systems for Manufacturing Eng., Mar. 9-13, Villeurbanne, France. Cosponsors: Assoc. Francaise pour la Cybernetique Economique et Technique et al. Contact INSA, 20 Av. Albert Einstein, 69621 Villeurbanne, France, phone 33 (72) 438-172, fax 33 (72) 440-800.

Ⓛ Fourth Int'l Conf. on Strategic Software Systems, Mar. 10-11, Huntsville, Ala. Cosponsors: IEEE Computer Soc. Huntsville Chapter, Univ. of Alabama in Huntsville, Contact Ann H. Yelle, Univ. of Alabama in Huntsville, Continuing Education Division, Conferences (SSS-92), Bevill Center 289-A, Huntsville, AL 35899, phone (800) 448-4035 or (205) 895-6372, fax (205) 895-6760.

Ⓛ Fifth Virus and Security Conf., Mar. 11-13, New York City. Sponsor: Data Processing Management Assoc. Financial Industries. Contact Judy S. Brand, PO Box 6313 FDR Station, New York, NY 10150, phone (800) 835-2246 ext. 190, fax (303) 825-9151.

Ⓛ EDAC 92, European Conf. on Design Automation, Mar. 16-19, Brussels. Cosponsors: EDAC Assoc. et al. Contact EDAC 92 Secretariat, CEP Consultants, 26-28 Albany St., Edinburgh EH1 3QH, UK, phone 44 (31) 557-2478, fax 44 (31) 557-5749.

Ⓛ Second Conf. on Computers, Free Dom, and Privacy, Mar. 18-20, Washington, DC. Cosponsors: ACM et al. Contact Lance Hoffman, George Washington Univ., Electrical Eng. and Computer Science Dept., Washington, DC 20052, phone (202) 994-4955.

Ⓛ IPPS 92, Sixth Int'l Parallel Processing Symp., Mar. 23-26, Beverly Hills, Calif. Contact Larry H. Canter, Computer Systems Approach, 1140 S. Raymond Ave., Suite B, Fullerton, CA 92631, phone (714) 738-3414, fax (714) 738-3470.



DCC '92 CALL FOR PAPERS



Data Compression Conference

(Sponsored by the IEEE Computer Society TCCC in Cooperation with NASA / CESDIS)

Snowbird, Utah

March 24 - 26, 1992

Program Committee:

R. Arps, *IBM*
 R. Baker, *PictureTel Inc.*
 A. Blumer, *Tufts U.*
 M. Cohn, *Brandeis U.*
 R. Capocelli, *U. Rome*
 R. Gallager, *MIT*
 R. Gray, *Stanford U.*
 D. Hirschberg, *UC Irvine*
 D. LeGall, *C-Cube Inc.*
 A. Lempel, *Technion*
 B. Lucier, *Purdue U.*
 J. Reif, *Duke U.*
 D. Sheinwald, *IBM*
 J. Storer, *Brandeis U.*
 J. Tilton, *NASA*
 V. Viswanathan, *TI Inc.*
 J. Vitter, *Brown U.*
 V. Wei, *Bellcore*
 I. Witten, *U. Calgary*

THEME: This conference will provide an international forum for current work on data compression and related areas. Topics of interest include but are not limited to: Coding theory, quantization theory, parallel compression algorithms and hardware, lossless and lossy compression algorithms for specific types of data (including text, images, video, speech, music, maps, instrument data, graphics, animation, and bit-maps), data compression standards, bi-level coding, transform methods, wavelet and fractal techniques, string searching and manipulation, closest-match retrieval, theory of minimal length encoding and applications to learning, system issues relating to data compression (including error control, data security, and indexing), medical imagery, scientific and space data archives.

SUBMISSION: Authors are requested to submit 4 copies of an extended abstract of 8 to 10 pages (*please include your affiliation, address, telephone number, fax number, and email address*) by November 15, 1991 to:

Prof. Martin Cohn
 Computer Science Dept.
 Brandeis University
 Waltham, MA 02254

(phone: 617-736-2705, fax: 617-736-2741, email: marty@cs.brandeis.edu)

NOTIFICATION: Authors will be notified of acceptance or rejection via email (or fax if necessary) on 12/13/92. A final camera-ready draft is due 1/13/92.

General Chair:

J. Storer, *Brandeis U.*

ADVANCE REGISTRATION: Registration includes receptions, banquet, coffee breaks, and proceedings. For registrations received before 2/15/92, the fee is \$295.00 for IEEE or affiliate members, \$370.00 for non-members, and \$150.00 for students; there is an additional \$100 late fee for all registrations after 2/15/92. If you have additional questions, call *Myrna Fox* at 617-736-2700.

Committee Chair:

M. Cohn, *Brandeis U.*

HOTEL: Space is limited; rooms may NOT be available after 2/15/92. Rooms are \$39 per person for a double room and \$78 for a single room. Call the Cliff Lodge, Snowbird, UT 84092, 800-453-3000 or 801-742-2222.

Publicity Chair:

R. Miller, *NASA*

TRANSPORTATION: The hotel may be reached by limousine or public bus from the Salt Lake City Airport.

REGISTRATION FORM

NAME: _____

AFFILIATION: _____

ADDRESS: _____

PHONE: _____ EMAIL: _____

☒ IEEE member ☐ yes ☐ no

AMOUNT ENCLOSED: _____

MAIL TO:

DCC Registration
 Computer Science Dept.
 Brandeis University
 Waltham, MA 02254

(Payment must be by check in U.S. dollars payable to DCC.)

BOOK REVIEWS

Editor: Alan Kaminsky, Department of Information Technology, Rochester Institute of Technology, PO Box 9887, Rochester, NY 14623-0887; (716) 475-5255; Internet ark@cs.rit.edu; Bitnet ark@cs.rit.edu

The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling

Raj Jain (Wiley-Interscience, New York, 1991, 576 pp., \$49.95)

Raj Jain's new book is an unusual object: a textbook to sit down and peruse. The prose is clear and fluent, but more important, it is witty. Wit is a good strategy for a textbook, because jokes based on what you've just learned reinforce both learning and enjoyment of the subject. The book is also practical. I read it while completing a performance evaluation project, and easily related its treatment of topics such as queue modeling to my hands-on work.

If you are doing a performance or systems engineering analysis for any kind of project, you will appreciate the list of "Reasons for Not Accepting the Results of Analysis." Here are some examples:

- It improves performance only for long IOs/packets/jobs/files, and most of the IOs/packets/jobs/files are short.
- It improves performance only for short IOs/packets/jobs/files, and who cares for the performance of short IOs/packets/jobs/files — it's the long ones that impact the system.
- It needs too much memory/CPU/bandwidth, and memory/CPU/bandwidth isn't free.
- It only saves us memory/CPU/bandwidth, and memory/CPU/bandwidth is cheap.
- This will violate the IEEE, ANSI, CCITT, or ISO standard.
- The standard says nothing about this, so it must not be important.

With rueful humor, Jain suggests using the list to show reviewers that all their arguments are old — or to help them critique an analysis that is competing with yours. I also imagined photocopying it to hand out at sys-

tems analysis meetings and cut hours off the discussion.

Jain works in the Distributed Systems and Performance Group at Digital Equipment Corporation. He specializes in networking, and has done important work on congestion management, Open Systems Interconnection protocols, traffic characterization, and Fiber Distributed Data Interface performance. The book grew out of a graduate course that Jain teaches at the Massachusetts Institute of Technology, and its examples come from projects carried out by Jain's students. This adds depth to the expository material, substantively illustrating difficult points with real-world experience from people who have used the text.

The major topics are measurement and workload techniques, probability and statistics, experimental design, simulation, and queueing models. An introductory section sets the tone by featuring mistakes often made in performance analysis. Jain dissects mistakes, myths, and games throughout the book. Some mistakes are honest; for instance, writing a benchmark program that runs a random set of I/O or network requests but doesn't accurately represent the system's performance for multiple requests in succession to one destination. Other mistakes are deliberate, and Jain anatomizes these in useful ways. Along these lines, he discusses myths about random number generation and common mistakes in simulation and experiment design. I do not want to give away punchlines, so I recommend reading the book.

Jain covers a lot of territory — both well-trodden and new — so it is inevitable that a few areas are not as fully

developed as the rest. For example, the chapter on monitoring techniques implies that few changes have occurred in the technology since the late 1970s. Distributed systems monitoring is presented only sketchily. The "implicit spying" technique (measurement by tapping a bus or network) is barely mentioned, and I would like to see more on issues such as clock synchronization and hybrid measurement approaches. Similarly, Jain gives a detailed description of remote terminal emulation, a technique that generates the workload of terminal users on one system and uses it to test another computer; but he makes only a passing observation about the changes that local area networking, workstations, and X Windows have made in the way people now use time-shared computers. It would be useful to get pointers on how to modify the old terminal emulation approach for the new environment.

The statistical and analytical topics are exceptionally lucid. The coverage is full enough to define network, operating systems, and systems engineering problems equally well. An introductory knowledge of probability theory and the motivation to work through examples are enough mathematical preparation for these. Jain has prepared fine summaries of the mathematical results, which are also set out in boxes for easy reference.

This book is a practical, rich, and enjoyable addition to the library of anyone whose work includes performance evaluation, simulation of systems, or system testing.

Allison Mankin
The Mitre Corporation

APPC: Introduction to LU 6.2

Alex Berson (McGraw-Hill, New York, 1990, 375 pp., \$49.95)

Imagine yourself in the neighborhood Computer BookLand store. You remember reading a review in *Com-*

puter of Alex Berson's book on a distributed processing protocol, but you don't remember any specific informa-

tion. You've scanned many shelves, but they are organized by categories like beginner, advanced, languages,

IBM, Apple, etc. You want to know for sure whether the book is there or not, so you approach the counter and present the details you have. The clerk keys this information in at a workstation. In a couple of seconds, the exact title comes up, and the clerk says, "We usually stock that book, but it's currently sold out. Would you like me to search other stores in the area?" You nod, and after a couple more keystrokes and a few seconds, the clerk smiles and informs you that their store a couple miles up the road has five copies.

This is an example of a full-blown Advanced Program-to-Program Communication application. When the local database search proved negative, the clerk connected to workstations in nearby Computer BookLand stores and executed remote programs capable of searching and reporting on other local databases. Depending on the design of the APPC application, some logic may be built into the search order (for example, closest stores first).

APPC is the application program interface to Logical Unit 6.2. APPC allows for distributed cooperative processing. LU 6.2 is converged in systems network architecture (SNA); it includes the protocols for application-subsystem-to-application-subsystem communication. Even though APPC

has been around since the mid-1980s, with IBM and many other vendors selling APPC platforms, Berson's is only the second commercially published book on APPC. One reason for this is the perception of APPC as too complicated for the consumer market. Unfortunately, this book does little to debunk this perception.

I found the book disappointing. This is not to say that it has no value. It introduces many important concepts, and includes some good explanations. It describes sessions, conversations, transaction programs, verbs, and flows. The explanation of establishing synchronization between programs is sound. The chapters on the Customer Information Control System data communication monitor are useful for people who have some background in CICS and SNA. The discussion of virtual telecommunication access method (VTAM) is reasonable, but contains that deadly phrase, "beyond the scope of this book."

The book is poorly edited and includes many grammatical and spelling errors. Moreover, it fails to address some important questions. Why does APPC have the functions that it has? Why are parallel sessions important? What about the SNA concept of "base and towers?" What about non-IBM platforms? There is no cogent expla-

nation of where and when to use APPC. The discussion of APPC implementations and future directions ends just when it becomes interesting.

The audience for this book ranges from the novice to those fairly advanced in SNA. Novices will find the book difficult; in fact, a beginner should read a primer on SNA before tackling this book. Intermediate to advanced readers may find the book tedious, but will also find some useful review and new information. I don't think the book is for experts.

Since 1990, when this book was published, many changes have occurred just in the IBM part of the APPC world. CICS and VTAM support for APPC has been enhanced. Implementations on additional platforms have been announced: the MVS operating system, the Information Management System (IMS) database data communication subsystem, and others. Advanced Peer-to-Peer Networking is now explicitly part of SNA. IBM Networking Services/2 for OS/2 workstations provides powerful APPC and APPN support. These changes have dated this book — all the more reason for a revised (and more carefully edited) edition.

Gary Shea
IBM Washington System Center

C/C++ for Expert Systems

David Hu (Management Information Source Inc., Portland, Ore., 1989, 565 pp., \$24.95)

If the contents of this book had hit the target as well as its timing, it could have been a best-seller. The expert systems application community is increasingly using C. While there are enough books on C, C++, and expert systems, a book combining these topics could contribute something new.

Unfortunately, the contents fall short. Hu claims to be writing for programmers who want to introduce expert system techniques into their systems and for knowledge engineers who want to convert existing expert systems into C and C++. However, in his early discussions of expert system basics and object-oriented programming, he treats some topics too superficially. He also chooses the unlikely vehicle of arithmetic operations to elucidate the advantages of an object-oriented approach. Inheritance is a crucial aspect of object-oriented systems, but Hu does not even mention in this example whether the number

classes, such as integers and real numbers, or their instances inherit relevant attributes or methods.

An appraisal of the strengths of C/C++ is well written, but the book does not begin to deliver on its promise — namely, to provide useful information about implementing expert systems in C/C++. The only helpful information in this regard concerns porting expert system software from Lisp or Prolog into C.

The overview of expert system tools is interesting but not entirely up to date (even for the publication date). The chapter on building an inference engine treats the usual subjects of backward and forward chaining and how to combine the two. The chapter on user interfaces does not discuss explanation components and how to support knowledge acquisition.

Library functions for expert system tools should have been treated in more depth. This subject touches the

real core of the work for implementing an expert system in C/C++. Hu only briefly mentions the library functions for object manipulation, reasoning, natural language, and file interaction.

The last part of the book comes closest to the mark. It surveys expert system types, addresses their respective limitations, and makes an interesting comparison of two well-known commercial systems, Xcon and Diesel Electric Locomotive Troubleshooting Aid. It also discusses the building and delivery of expert systems.

Of the five appendices, only the one on expert systems tools and the glossary are worthwhile. In short, since this book offers little that is not handled better elsewhere, it cannot be recommended.

Claudia Johnson
Siemens Aktiengesellschaft
Munich, Germany

**POSTDOCTORAL FELLOW
POSITIONS**
UNIVERSITY OF MINNESOTA
Army High Performance Computing
Research Center
(AHPCRC)

The AHPCRC runs an interdisciplinary research program that focuses on the high performance computational aspects of engineering, physical sciences, computer science and mathematics. The primary computational system is a 32K Connection Machine, CM-2 with 1,024 64-bit Vector Floating Point units and 4 gigabytes of main memory. Other computational resources include a Graphics and Visualization Laboratory and access to a Cray-2/4-512, a Cray X-MP-EA/4-64, and a Cray X-MP-EA/2-64. The Center is currently emphasizing research activities in four focus areas:

Computational Continuum Mechanics, including computational fluid dynamics and solid mechanics with finite volume and finite element techniques;

High Performance Computing Issues in Nonlinear Dynamics, with special emphasis on long-time dynamical issues arising in fluid flows, convective flows, and reacting flows;

Computer Science, including parallel algorithms for matrix computations and optimization, systems environment tools, and performance evaluations.

Emerging Focus Areas, covering computational biomedical sciences, computational chemistry, and modern materials.

We are inviting applications for post-doctoral fellowships (research associate positions) beginning in September 1992, but alternative starting dates may be requested. All requirements for a doctorate must be completed prior to the starting date. The materials listed below must be received by January 15, 1992:

1. Description (maximum two pages) of research background and plans for participating in one of the AHPCRC focus areas. (A description of project opportunities in these focus areas is available from the AHPCRC; please refer to "Research Opportunities for AHPCRC Postdoctoral Fellows.")
2. Curriculum vitae including a list of publications.
3. Three letters of recommendation, to be sent directly to the AHPCRC.

All correspondence should be directed to:

Toyfun E. Tezduyar,
Director of Fellows Program
AHPCRC/University of Minnesota
1100 Washington Avenue South
Minneapolis, Minnesota 55415

The University of Minnesota is an equal opportunity
educator and employer.

THE OPEN CHANNEL

Editor: Will Tracz, IBM, Federal Sector Division, Mail Drop Q210, Owego, NY 13827
e-mail tracz@Owgv001.nus1.ibm.com

Confessions of a used-program salesman: "Not again!"

Lately I have been both encouraged and discouraged by the surge of interest in used-program sales and service.

Encouraged, for instance, by requests like the one from Barry Boehm, director of the DARPA's Software and Intelligent Systems Technology Office. He has issued the "Megaprogramming Challenge" to researchers, calling on them to develop the technology for constructing software "one component at a time rather than one line of code at a time."

This adds some credibility to the used-program business, which is booming. It seems like every time I turn around, a new dealership is popping up. Misery loves company, I always say. Not that being a used-program salesperson isn't fun. It is, and I really do enjoy the competition. It's just that the owners of these new establishments may not really be prepared for this line of business. They don't seem to practice what they preach, and I fear they are doomed to fall into the same potholes that caused other used-program dealerships to crash and burn.

In graduate school I learned that an hour in the library can save a month in the laboratory. This lesson applies also to a library of used programs that could save valuable time designing, coding, documenting, and testing new programs. But this is not the turkey I want to roast.

I want to focus on "advances" in the state of the art in software reuse. Doug McIlroy first proposed used-program sales in 1968 at the NATO Conference on Software Engineering. My suspicions on (the lack of) progress in the field since then were confirmed recently by Peter Freeman. In his keynote address to the First International Workshop on Software Reuse, July 1991, in Dortmund, Germany, Peter observed that software reuse researchers were "stepping on each other's toes, rather than standing on each other's shoulders." He suggested that instead of 23 years of progress in advancing the state of the art in software reuse, researchers had repeated one year of progress 23 times.

But then again, why should we be any different from anyone else? I have

often heard that new technology needs to be invented three times before it catches on. I guess that in our case, we just have to reinvent it three times.

Will Tracz
IBM Federal Sector Division
Owego, N.Y.

Let's reinvent the wheel!

Have you noticed how the computer community is going head over heels for artificial intelligence? Some people claim that AI aims at making computers think like the human brain. And therein hangs a tale. Has anyone ever stood up to ask, "Why mimic human intelligence at all?"

Let's put aside this question for a moment and go back a few decades when engineers started designing the first automobiles. Imagine the plight of our cars, if these pioneers had started working on "artificial legs" to amputate their vehicles! Or imagine early airplane designers working on "artificial wings" to propel us across the skies — no engines, no jets, no propellers, just wings that flap like those of birds and butterflies! Daedalus and his likes have not persisted, thank heaven.

Imagine a world where God had made people and butterflies with wheels, cogs, and pulleys! Have you ever seen an animal in nature moving on wheels? Sounds ridiculous, doesn't it? A wheel is a wheel and a leg is a leg; each has its own *raison d'être*. There is little in common between the two, yet each fits beautifully in its place.

The moral of this story is simple and clear: Let us AI folks invent an intellectual wheel. Never mind if this "wheel" cannot think like the human brain. It will still work wonders for humanity, just like the mechanical wheel. Let's leave something worthwhile for our children and grandchildren.

S. Parthasarathy
CMC Limited
Secunderabad, India

Preliminary CALL for PAPERS

OOPSLA^{'92}

Seventh Annual ACM Conference on

Object-Oriented Programming Systems, Languages, and Applications

18-22 October 1992, Vancouver, British Columbia, Canada

Program Committee

Rebecca Wirfs-Brock, Tektronix (chair)
Sam Adams, Knowledge Systems
Brian Barry, Object Technology
Grady Booch, Rational
Timothy Budd, Oregon State U.
Pierre Cointe, Rank Xerox
William Cook, Apple Computer
Ward Cunningham, Wyatt Software
Richard Helm, IBM T.J. Watson
Ivar Jacobson, Objective Systems
Mark Linton, Silicon Graphics
Steven D. Litvintchouk, MITRE
Ole Lehrmann Madsen, Aarhus U.
John C. Mitchell, Stanford
Eliot Moss, U. Massachusetts
Patrick O'Brien, DEC
James R. Rumbaugh, GE
Alan Snyder, Sun Microsystems
Joshua Sussner, Apple Computer
Mario Tokoro, Keio U. / Sony CSL
Chris Tomlinson, MCC
Dionysios Tsihrizitis, U. Geneva
Daniel Weinreb, Object Design
K.C. Burgess Yakemovic, NCR

Program Chair

Rebecca Wirfs-Brock
OOPSLA '92 Program Chair
correspondence:
Tektronix
M/S 47-786
P.O. Box 500
Beaverton OR 97077
+1-503-627-3045
rebecca@tekig1.tek.com
submissions:
16200 SW Pacific Hwy
Suite 187
Tigard, Oregon 97224

Conference Chair

John Pugh
OOPSLA '92 Conference Chair
School of Computer Science
Carleton University
Ottawa, Ont.
Canada K1S 5B6
+1-613-788-4330
oopsla92@carleton.ca

The annual OOPSLA conference is the premier forum bringing together participants in the field of object technology to share ideas and experiences. The combination of research and experience papers brings together both academic and industrial participants and ensures a rich exchange of ideas. Areas of interest include but are not limited to: language design and implementation, tools and environments, components and frameworks, user interfaces, principles and theory, concurrent and distributed systems, databases and persistence, design methods and software engineering practices

OOPSLA provides a unique opportunity to share your research and experience with other leaders in these areas of software development.

Papers

The conference will include both invited and contributed papers. Authors are encouraged to submit high quality papers describing relevant research or experience. Research papers should describe work whose purpose is to advance the state of the art of object technology. Experience papers should describe the practical application of object-oriented methods. The program committee will evaluate each paper on its relevance, clarity, correctness, originality, and significance. Special consideration will be given to promising experience papers. Papers must not be under consideration elsewhere (or published) in the same or similar form. Do not submit multiple papers describing essentially the same work.

Authors should send six copies of the full paper, in English, to the program chair to be received no later than **1 March 1992**. Papers must be limited to 18 pages, typed double spaced. Each copy must contain contact information (contact name, postal address, and phone number), a 100-word abstract, and indicate the paper category (research or experience). *Submissions failing to meet these conditions will be rejected.*

Authors will be notified of acceptance or rejection by **20 May 1992**. Camera-ready copies of accepted papers are due **20 July 1992**. Authors of accepted papers are expected to sign an ACM copyright release form and present the paper at the conference. Proceedings will be distributed at the conference and via *SIGPLAN Notices* and will be available from ACM Press. Outstanding papers may be considered for a special issue of a journal.

Guidelines for authors will be published in the proceedings of the OOPSLA '91 conference, or can be obtained from the program chair.

Experience Reports

Space will be made available in a separate track for short presentations of unrefereed reports describing

practical experience applying object technology to production quality software development. Prospective speakers should submit a 1-2 page description of the project scope and status and the specific points to be covered in the presentation to the program chair by **1 April 1992**. Selection will be based on relevance and potential interest. Summaries will be collected after the conference for publication in *OOPS Messenger*.

Other Contributions

For information on tutorials, panels, workshops, demos, poster papers, and vendor exhibits, see the Call for Participation, which will be available in October at the 1991 conference or from the conference chair.

Important Dates

1 March 1992
Papers due
1 April 1992
Experience reports due
20 May 1992
Acceptance notification
20 July 1992
Camera-ready papers due



SIGPLAN

Papers due 1 March 1992

**"If only we could use UNIX® for
our realtime applications..."**



You want the power of a standard development OS. But you also want the performance of a realtime OS. Which way do you turn?

Presenting QNX 4.0. The operating system that's responsive enough for realtime apps, small enough for PC platforms, flexible enough for transparent networking, and modular enough for the most demanding configurations.

POSIX Means Portable

Operating systems come in more flavors than ice cream. Which is why IEEE's POSIX standard is now such an important safeguard of portability.

QNX is a high-performance realtime OS with its own unique microkernel architecture.* But its API is based on the latest POSIX standards, so you get both outstanding performance and portability for all your apps.

Performance At Run Time, Design Time, All the Time

Only QNX combines the performance of a dedicated realtime executive with the time-saving benefits of a rich development environment—including a host of utilities, an award-winning C compiler, and an optional OPEN LOOK® GUI package.

QNX Is Distributed

The QNX operating system lets you extend the limits of any one microprocessor. Whether you're running a network of four or 400 machines, QNX makes it all feel like a single computer.



Interprocess communication is network-wide, so every process can transparently access every resource—programs, files, devices, even CPUs—anywhere on the network. And you can set up your network using any mix of Intel-based PCs.

Responsive Tech Support

QNX's support hotline can put you in direct contact with the Technical Development team itself. And you'll have access to our 24-hour online conferencing and update system, where the response time to your questions is almost like real time.

"If Only..."

If you think you have to choose between a standard development OS and a high-performance realtime OS, think again.

Now you can have it both ways.

**For more information, phone
1-800-363-9001.**

Quantum Software Systems Ltd. • 175 Terrence Matthews Crescent • Kanata, Ontario, Canada • K2M 1W8 • 1-613-591-0931

[A 16 microsecond preemptive context switch on a 33 MHz 486 (100 microseconds on a 16 MHz 286)]

*QNX is not based on UNIX source code.

QNX is a registered trademark of Quantum Software Systems Ltd. UNIX and OPEN LOOK are registered trademarks of UNIX System Laboratories, Inc. Intel is a registered trademark of Intel Corporation. © Quantum Software Systems Ltd. 1991

Reader Service Number 3